

Entwicklung paralleler Algorithmen zur numerischen Simulation von Gas–Partikel–Strömungen unter Berücksichtigung von Partikel–Partikel–Kollisionen

Dipl.–Ing. Erik Wassen

Chemnitz, Dezember 1998

Von der Fakultät für Maschinenbau und Verfahrenstechnik der Technischen
Universität Chemnitz genehmigte Dissertation

Datum der Einreichung: 15.09.1998

Datum der Verteidigung: 14.12.1998

Gutachter: Prof. Dr. H. Herwig
Prof. Dr. N. Aksel
Prof. Dr. F. Ebert

meinen Eltern

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter an der Technischen Universität Chemnitz in der Forschungsgruppe Mehrphasenströmungen der Professur Technische Thermodynamik. Der Großteil der hier dokumentierten Forschungsarbeiten wurde im Rahmen des Teilprojektes D2 des Sonderforschungsbereiches 393 “Numerische Simulation auf massiv parallelen Rechnern“ durchgeführt.

Mein besonderer Dank gilt Herrn Prof. Dr. H. Herwig, dem Leiter der Professur Technische Thermodynamik, für seine Bereitschaft, diese Arbeit zu betreuen, und insbesondere für sein großes persönliches Engagement bei der Durchführung dieses Promotionsvorhabens. Des weiteren danke ich Herrn Prof. Dr. N. Aksel und Herrn Prof. Dr. F. Ebert für die Übernahme der Begutachtung der Dissertation.

Darüber hinaus danke ich den Kollegen der Forschungsgruppe Mehrphasenströmungen, insbesondere Herrn Dr. Thomas Frank, dem Leiter der Numerik-Gruppe, und Herrn Dr. Qiang Yu, für die fachlichen Diskussionen und Ratschläge sowie für das überaus angenehme Arbeitsklima in der Gruppe.

Der überwiegende Teil der in dieser Arbeit beschriebenen Simulationen wurde während meines zweimonatigen Aufenthaltes am Edinburgh Parallel Computing Centre (EPCC) in Edinburgh, Schottland, durchgeführt. Dieser Forschungsaufenthalt erfolgte im Rahmen des Programmes “Training and Research on Advanced Computing Systems“ (TRACS) der Europäischen Union. Bei den Mitarbeitern des EPCC möchte ich mich für die logistische Unterstützung und die großzügige Bereitstellung von Rechenzeit bedanken.

Erik Wassen, Dezember 1998

Inhaltsverzeichnis

1	Einleitung	1
2	Berechnung der Gasströmung	4
2.1	Mathematisches Modell für die Gasströmung	4
2.1.1	Grundgleichungen	4
2.1.2	Koordinatensystem	7
2.1.3	Randbedingungen	9
2.2	Numerische Lösung der Bilanzgleichungen	10
2.2.1	Gitter für die numerische Simulation	10
2.2.2	Diskretisierung der Bilanzgleichungen nach der Finite-Volumen-Methode	11
2.2.3	Lösungsmethode	14
3	Berechnung der Bewegung der Partikelphase	16
3.1	Parameter zur Klassifizierung von Gas-Partikel-Strömungen	16
3.2	Das Euler/Euler- oder Zwei-Fluid-Modell	19
3.3	Das Euler/Lagrange-Verfahren	20
3.3.1	Berechnung der Bewegung eines Einzelpartikels	21
3.3.2	Berechnung der Strömung der dispersen Phase	36

4	Simulation von Partikel–Partikel–Kollisionen	39
4.1	Bedeutung der Partikel–Partikel–Kollisionen	39
4.2	Allgemeine Definitionen und Begriffe	41
4.3	Berechnung der Kollision zweier Partikel	42
4.4	Modelle für die Simulation der Kollisionen	45
4.4.1	Trajektorienbasierte Kollisionsmodelle	45
4.4.2	Kollisionsmodelle auf der Basis der simultanen Partikelverfolgung	50
5	Algorithmen für die parallele Simulation	59
5.1	Motivation und Allgemeines	59
5.2	Parameter zur Bewertung paralleler Algorithmen	65
5.3	Parallele Berechnung der Gasströmung	66
5.4	Parallele Berechnung der kollisionsbehafteten Partikelbewegung	69
5.4.1	Stand der Forschung	69
5.4.2	Neue parallele Algorithmen für die trajektorienbasierte Simulation	70
5.4.3	Neuer paralleler Algorithmus für die Simulation auf der Basis der simultanen Partikelverfolgung	80
6	Ergebnisse der Testrechnungen	84
6.1	Berechnung der Strömung in einem vertikalen Kanal	85
6.1.1	Physikalische Eigenschaften der Strömung	85
6.1.2	Anwendung der parallelen Simulationsalgorithmen auf die Berechnung der vertikalen Kanalströmung	93
6.2	Berechnung der Strömung in einem horizontalen Kanal	105
6.2.1	Physikalische Eigenschaften der Strömung	105
6.2.2	Anwendung der parallelen Simulationsalgorithmen auf die Berechnung der horizontalen Kanalströmung	112

7	Schlußfolgerungen zur Anwendbarkeit der parallelen Algorithmen	121
7.1	Allgemeines	121
7.2	Parallele Trajektorienberechnung	122
7.3	Parallele simultane Partikelverfolgung	124
8	Zusammenfassung und Ausblick	126
A	Erläuterungen zur Diskretisierung und Lösung der Erhaltungsgleichungen für die Gasströmung	130
A.1	Approximation der diffusiven Flüsse	130
A.2	Berechnung der Ableitung im Zellenmittelpunkt	132
A.3	Lösung des linearen Gleichungssystems für eine Strömungsvariable mit der SIP-Methode	132

Verzeichnis der Formelzeichen

Formelzeichen	Bedeutung
$\overline{\overline{D}}$	Deformationstensor
$C_\mu, C_{\varepsilon 1}, C_{\varepsilon 2}$	Konstanten für das k - ε -Modell
DF_k, DF_ε	diffusiver Transport von k bzw. ε
DS_k, DS_ε	Dissipation von k bzw. ε
E	Effizienz
F	allgemeiner Fluß
F^c, F^d	konvektiver bzw. diffusiver Fluß
\vec{F}_C	Kraft auf ein Partikel während der Kollision
\vec{F}_G	Gravitationskraft
\vec{F}_M	Magnus-Kraft
\vec{F}_S	Saffman-Kraft
\vec{F}_W	aerodynamische Widerstandskraft
H_r	mittlere Rauigkeitstiefe
$\overline{\overline{I}}$	Einheitstensor
I_P	Trägheitsmoment des Partikels
\vec{J}	Kraftimpuls
L_E	Durchmesser eines Turbulenzwirbels
$L_{P,E}$	Abstand zwischen den Punkten P und E
L_r	mittlerer Abstand zwischen Rauigkeitsspitzen
\dot{N}_P	Partikelstrom entlang einer Trajektorie
N_T	Anzahl von Trajektorien
N_{real}, N_{sim}	Anzahl realer bzw. simulierter Partikel
P_C	Kollisionswahrscheinlichkeit
P_k, P_ε	Produktion von k bzw. ε
\vec{Q}	allgemeiner Quelltermvektor
$\vec{Q}_{P\vec{v}}, Q_{Pu}, Q_{Pv}$	Partikel-Quellterme für die Impulsgleichungen
Re_P	Partikel-Reynoldszahl
Re_S	Reynoldszahl der Scherströmung
Re_ω	Reynoldszahl der Rotation
S	Oberfläche eines Kontrollvolumens
\vec{S}, S_x, S_y	Flächenvektor

Formelzeichen	Bedeutung
Sp	Speedup
St	Stokes–Zahl
$\overline{\overline{T}}$	Spannungstensor
\vec{T}	Drehmoment
T_D	Durchgangszeit durch einen Turbulenzwirbel
T_E	Wirbellebendauer
T_{Komm}	Kommunikationszeit
T_N	Rechenzeit für parallele Rechnung auf N Prozessoren
$T_{R,max}$	maximale Rechenzeit eines Nodes bei paralleler Rechnung
T_1	Rechenzeit für serielle Rechnung
c_M	Beiwert für die Magnus–Kraft
c_S	Beiwert der Saffman–Kraft
c_T	Konstante für das LSD–Modell
c_W	Widerstandsbeiwert
c_ω	Rotationsbeiwert
d_P	Partikeldurchmesser
d_{P0}	mittlerer lokaler Partikeldurchmesser
e_p	Stoßzahl für den Partikel–Partikel–Stoß
e_w	Stoßzahl für den Partikel–Wand–Stoß
f	allgemeiner Fluß pro Flächeneinheit
$\vec{f}_{PF}, f_{PFx}, f_{PFy}$	von Partikeln auf das Fluid pro Volumeneinheit ausgeübte Kraft
f_p	Reibungsbeiwert für den Partikel–Partikel–Stoß
f_w	Reibungsbeiwert für den Partikel–Wand–Stoß
\vec{g}, g_x, g_y	Gravitationsbeschleunigung
\vec{i}_ξ	Richtungsvektor
k	kinetische Turbulenzenergie
\dot{m}_F, \dot{m}_P	Fluid– bzw. Partikelmassenstrom
m_P	Masse eines Partikels
\vec{n}, n_x, n_y	Normalenvektor
n_P	lokale Anzahldichte der Partikel
\vec{n}_v	Normalenvektor der virtuellen Wand
n_{P0}	gemittelte Anzahldichte der Partikel
p	Druck

Formelzeichen	Bedeutung
q	allgemeine Variable
\vec{r}	Ortsvektor
t	Zeit
Δt	Zeitschritt
Δt_w	Wechselwirkungszeit des Partikels mit einem Turbulenzwirbel
u'_F, v'_F	Schwankungsanteile der Fluidgeschwindigkeit
\vec{v}_F, u_F, v_F	zeitlich gemittelte Fluidgeschwindigkeit
\vec{v}_P, u_P, v_P	Partikelgeschwindigkeit
$\vec{v}_{P0}, u_{P0}, v_{P0}$	mittlere lokale Partikelgeschwindigkeit
\vec{v}_{fc}	Relativgeschwindigkeit der Partikeloberflächen beim Stoß
v_r	Relativgeschwindigkeit zwischen Partikeloberfläche und Wand
\vec{v}_{rel}	Relativgeschwindigkeit zwischen Partikel und Fluid
$\vec{v}_{rel,c}$	Relativgeschwindigkeit zweier kollidierender Partikel
x, y	kartesische Koordinaten
Γ	Diffusionskoeffizient
Ω	Größe eines beliebigen Kontrollvolumens
α_F, α_P	Volumenanteil des Fluides bzw. der Partikel
β	Wichtungsfaktor
γ	Inklination der virtuellen Wand
$\Delta\gamma$	Standardabweichung von γ
δ_P	mittlerer Partikelabstand
ε	Dissipationsrate
η	Massenbeladung
κ	von Karmansche Konstante
λ_P	mittlere freie Weglänge
λ_e	Interpolationsfaktor
μ_F	effektive dynamische Viskosität des Fluides
μ, μ_t	molekulare und scheinbare Viskosität
ν_C	Kollisionsfrequenz
ν_F	kinematische Viskosität des Fluides
ρ_F, ρ_P	Materialdichte des Fluides bzw. des Partikels
σ_F, σ_P	Gemischdichte der Fluid- bzw. Partikelphase
σ_M	Konstante zur Bestimmung von c_M

Formelzeichen	Bedeutung
σ_T	Kollisionsquerschnitt
$\sigma_k, \sigma_\varepsilon$	Konstanten für das k - ε -Modell
τ_A	aerodynamische Relaxationszeit
τ_C	mittlere Kollisionszeit
τ_S	charakteristische Systemzeit
τ_{xx}, τ_{xy}	Komponenten des Spannungstensors
ϕ	zeitlicher Mittelwert einer allgemeinen Strömungsgröße
ϕ'	Schwankungswert einer allgemeinen Strömungsgröße
ϕ_{akt}	Momentanwert einer allgemeinen Strömungsgröße
ϕ_e, ϕ_E, ϕ_P	Wert von ϕ am Ort e , E bzw. P
ψ	Zufallszahl
$\vec{\omega}_F, \omega_{Fx}, \omega_{Fy}$	Rotation des Fluides
$\vec{\omega}_P, \omega_{Px}, \omega_{Py}$	Rotationsgeschwindigkeit des Partikels
$\vec{\omega}_{P0}, \omega_{Px0}, \omega_{Py0}$	mittlere lokale Rotationsgeschwindigkeit der Partikel
ω_{rel}	Rotation des Partikels relativ zum Fluid

Kapitel 1

Einleitung

Mehrphasige Strömungen finden sich in weiten Teilen der Energie- und Verfahrenstechnik, der Umwelttechnik sowie der Lebensmittel- und auch der pharmazeutischen Industrie. Je nach Aggregatzustand und Volumenanteil der beteiligten Phasen kann der Charakter dieser Strömungen sehr unterschiedlich sein. Beispiele für zweiphasige Gas-Flüssigkeits-Strömungen sind blasenhaltige Flüssigkeiten, wie sie oft in Reaktoren in der Verfahrens- bzw. Bioverfahrenstechnik vorkommen, oder auch tropfenhaltige Gasströmungen, wie sie z.B. bei der Verbrennung flüssiger Brennstoffe oder bei sonstigen Sprühvorgängen auftreten. Ein Beispiel für Gas-Feststoff-Strömungen ist der weithin gebräuchliche Transport granularen Materials durch Rohrleitungen mit Hilfe eines Trägergases (pneumatischer Transport). Im Bereich der ölfördernden bzw. -verarbeitenden Industrie finden sich z.B. auch dreiphasige Strömungen mit Anteilen von Gas, Wasser und Öl.

Betrachtet man Gemische, in denen eine Phase ein Gas oder eine Flüssigkeit ist und somit in kontinuierlicher Form vorliegt, und die übrigen Anteile in Form diskreter Teilchen vorliegen, so bezeichnet man diese Gemische als dispers. In dieser Arbeit sollen ausschließlich Gas-Feststoff-Strömungen behandelt werden, bei denen also die kontinuierliche Phase ein Gas ist und die disperse Phase aus festen Partikeln besteht.

Anlagen bzw. Apparate, in denen Gas-Partikel-Strömungen zu finden sind, können sehr unterschiedliche Aufgaben erfüllen. Betrachtet man z.B. Kraftwerke, die mit Kohlestaub befeuert werden, so wird dort die Kohle nach ihrer Zerkleinerung in Mühlen mit Hilfe des pneumatischen Transportes durch ein Rohrleitungssystem zum Brennraum transportiert. Hierbei ist es wichtig, daß trotz der in den Rohrleitungen vorhandenen Krümmern und Verzweigungen sämtliche Zuleitungen zum Brennraum denselben Brennstoffmassenstrom liefern. Die Injektion des Luft/Kohle-Gemisches in den Brennraum erfolgt mit Hilfe von Düsen, die eine möglichst gute Durchmi-

schung der beiden Phasen gewährleisten müssen. Das Abgas des Verbrennungsvorgangs enthält Aschepartikel, die vor dem Entweichen des Abgases in die Umgebung durch einen Separationsprozeß, z.B. innerhalb eines Zyklons, aus dem Gas entfernt werden müssen.

Die Voraussetzung für die optimale Auslegung solcher Mischungs-, Separations- und Transportprozesse ist eine genaue Kenntnis der physikalischen Eigenschaften der entsprechenden Gas-Feststoff-Strömung. Bei der Untersuchung dieser grundlegenden Eigenschaften sind in den letzten Jahren sowohl auf dem Gebiet der Meßtechnik als auch auf dem Gebiet der Modellierung und der numerischen Simulation große Fortschritte erzielt worden. Betrachtet man den Bereich der numerischen Simulation, so hat sich das Lagrange-Verfahren für die Berechnung der Bewegung der dispersen Phase als besonders geeignet erwiesen.

Viele Untersuchungen der vergangenen Jahre haben gezeigt, daß die Partikel-Partikel-Kollisionen in Gas-Feststoff-Strömungen bereits bei moderaten Partikelbelastungen einen großen Einfluß auf das Strömungsverhalten haben. Darauf aufbauend wurden verschiedene Ansätze entwickelt, um diese Kollisionen innerhalb des Lagrangeschen Berechnungsverfahrens zu berücksichtigen. Dadurch wurde eine exaktere Modellierung der physikalischen Strömungseigenschaften erreicht. Andererseits jedoch erhöht sich durch die Einbeziehung der Kollisionen der Rechenaufwand für eine Simulation deutlich. Die numerische Berechnung disperser Mehrphasenströmungen mit Hilfe des Lagrange-Verfahrens ist heutzutage mit zu den Anwendungen im Bereich der Strömungssimulation zu zählen, die die höchsten Anforderungen an die Rechnerressourcen stellen.

Wenn man andererseits die Entwicklung der Rechentechnik in den vergangenen Jahren betrachtet, so ist auch hier ein ständiger Fortschritt zu verzeichnen. Die derzeit leistungsfähigsten Rechner sind ohne Ausnahme massiv parallele Systeme, die aus bis zu mehreren tausend Prozessoren aufgebaut sind. Um das große Leistungspotential dieser Computer effizient nutzen zu können, müssen die verwendeten Berechnungsalgorithmen an die besondere Architektur der Parallelrechner angepaßt werden. Für den Fall der parallelen Simulation einphasiger Strömungen existiert eine Vielzahl von Forschungsarbeiten, die sich mit unterschiedlichen Aspekten der Parallelisierung befassen. Auf dem Gebiet der parallelen Berechnung disperser Mehrphasenströmungen sind jedoch erst sehr wenige Arbeiten bekannt.

Es ist deshalb das Ziel der vorliegenden Arbeit, Algorithmen zu entwickeln, die die numerische Simulation kollisionsbehafteter Gas-Partikel-Strömungen auf massiv parallelen Rechnern ermöglichen. Dadurch kann die Rechenzeit, die für solche Simulationen benötigt wird, signifikant verkürzt werden. Die Effizienz der parallelen Algorithmen soll anhand verschiedener Testfälle untersucht werden. Auf der Grundlage dieser Ergebnisse können Ansätze für eine weitere Verbesserung der Al-

gorithmen sowie für alternative Parallelisierungsverfahren entwickelt werden.

Betrachtet man allgemein die Aufgabe “Entwicklung eines Algorithmus“, so sind zur Lösung dieser Aufgabe drei wesentliche Arbeitsschritte erforderlich: 1) Das Finden eines Algorithmus, der potentiell die gewünschten Anforderungen erfüllt, auf theoretischem Wege, 2) die Umsetzung dieses Algorithmus in einen Programmcode mit Hilfe einer höheren Programmiersprache und 3) das Testen des in einem Programm implementierten Algorithmus anhand geeigneter Testfälle. Als vierter Schritt kommt gegebenenfalls noch die Optimierung des Algorithmus auf der Grundlage der Testergebnisse hinzu. Der weitaus zeitaufwendigste Teil der Entwicklungsarbeit ist der zweite Schritt, d.h. die programmtechnische Realisierung des theoretisch gefundenen Algorithmus. Dieser Arbeitsschritt bildet einen Anteil von ca. 70–80% am Gesamtzeitaufwand für die Algorithmenentwicklung. Die dabei auftretenden Probleme sind jedoch meist relativ spezieller Natur, z.B. in Abhängigkeit von der verwendeten Programmiersprache oder des eingesetzten Rechners. In der vorliegenden Arbeit soll deshalb die Beschreibung der übergeordneten Algorithmen im Vordergrund stehen, deren Anwendbarkeit und Leistungsmerkmale von diesen Faktoren unabhängig sind. Es sei jedoch an dieser Stelle angemerkt, daß die Implementierung eines parallelen Programmes eine deutlich komplexere Aufgabe darstellt als die Implementierung eines Programmes auf einem seriellen Rechner, da im parallelen Fall dasselbe Programm gleichzeitig auf einer großen Anzahl separater Prozessoren ausgeführt wird.

Im Kapitel 2 wird zunächst kurz das numerische Verfahren beschrieben, das hier zur Simulation der Gasströmung verwendet wird. Anschließend wird im Kapitel 3 die Modellierung und Berechnung der Partikelbewegung dargestellt. Die verschiedenen Modelle, die zur Berücksichtigung von Partikel–Partikel–Kollisionen innerhalb des Lagrange–Verfahrens existieren, werden im Kapitel 4 vorgestellt. Im Kapitel 5 erfolgt dann die Beschreibung der parallelen Berechnungsverfahren. Dabei wird zunächst die parallele Berechnung der Gasströmung beschrieben. Insbesondere aber werden in diesem Kapitel die neuen parallelen Algorithmen für die Berechnung der Partikelbewegung vorgestellt, die im Rahmen dieser Arbeit entwickelt wurden. Zur Untersuchung der Algorithmen wurden diese auf die Berechnung zweier Testströmungen angewendet, einer vertikalen und einer horizontalen Kanalströmung. Die Ergebnisse dieser Untersuchungen sind im Kapitel 6 dargestellt. Die sich aus diesen Ergebnissen ergebenden Schlußfolgerungen und weiteren Entwicklungsmöglichkeiten sind schließlich im Kapitel 7 beschrieben.

Kapitel 2

Berechnung der Gasströmung

Bei den in dieser Arbeit betrachteten dispersen Mehrphasenströmungen ist die fluide Phase ein Gas, und die disperse Phase besteht aus festen Partikeln. In diesem Kapitel soll das Verfahren dargestellt werden, mit dem die Bewegung der Gasphase berechnet wird. Dazu werden zunächst im Abschnitt 2.1 die Eigenschaften der hier betrachteten Gasströmungen beschrieben und die Grundgleichungen angegeben, die die Bewegung des Gases beschreiben. In dem folgenden Abschnitt 2.2 wird das Verfahren zur numerischen Lösung dieser Gleichungen angegeben, wobei auf die Art der Diskretisierung, die Wahl des numerischen Gitters sowie das Lösungsverfahren für die diskretisierten Gleichungen eingegangen wird.

2.1 Mathematisches Modell für die Gasströmung

2.1.1 Grundgleichungen

Es werden in dieser Arbeit ausschließlich inkompressible, turbulente und statistisch stationäre Gasströmungen betrachtet. Der aktuelle Wert ϕ_{akt} einer allgemeinen Strömungsgröße ϕ in einer turbulenten Strömung läßt sich durch die Summe aus einem zeitlichen Mittelwert und einer stochastischen Schwankungsgröße darstellen:

$$\phi_{akt}(\vec{r}, t) = \phi(\vec{r}) + \phi'(\vec{r}, t) . \quad (2.1)$$

Hierbei bezeichnet \vec{r} den Ort, an dem die Größe betrachtet wird, t ist die Zeit, $\phi(\vec{r})$ ist der zeitlich gemittelte Wert, und $\phi'(\vec{r}, t)$ ist der Schwankungswert. Alle im folgenden angegebenen Variablen für die Strömungsgrößen bezeichnen, sofern sie nicht besonders gekennzeichnet sind, immer die zeitlich gemittelten Werte dieser Größen.

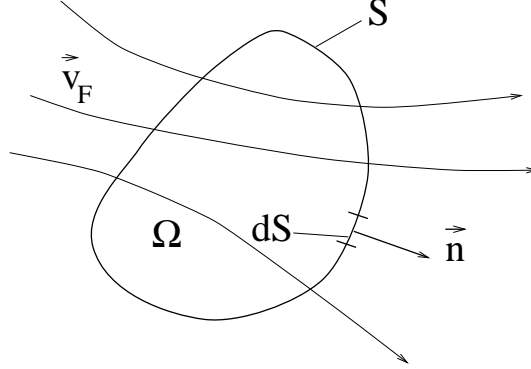


Abbildung 2.1: Allgemeines durchströmtes Kontrollvolumen

Die Bewegung des Fluides wird durch die Bilanzgleichungen für Masse und Impuls beschrieben. Außerdem sind gegebenenfalls weitere Gleichungen für verschiedene skalare Größen erforderlich. Die Bilanzgleichungen können auf viele unterschiedliche Arten formuliert werden. Nachfolgend wird die Integralform gewählt, da diese der Ausgangspunkt für das hier verwendete numerische Lösungsverfahren ist. Man erhält die Integralform, wenn man die Bilanzgleichungen für ein beliebig geformtes, nicht bewegtes Kontrollvolumen Ω mit der Oberfläche S formuliert, wie es in Abbildung 2.1 dargestellt ist.

Unter den oben genannten Voraussetzungen ergibt sich die Kontinuitätsgleichung, d.h. die Massenerhaltungsgleichung, zu:

$$\int_S \rho_F \vec{v}_F \cdot \vec{n} \, dS = 0 , \quad (2.2)$$

wobei ρ_F die Dichte des Fluides, \vec{v}_F die Fluidgeschwindigkeit, dS ein infinitesimal kleines Element der Oberfläche S des Kontrollvolumens und \vec{n} der zu dS gehörende, nach außen gerichtete Normalenvektor ist. Bei den in dieser Arbeit betrachteten Mehrphasenströmungen finden keine Phasenübergänge statt, weshalb die Kontinuitätsgleichung keine Quellterme enthält. Betrachtete man dagegen Strömungen mit verbrennenden, verdampfenden oder kondensierenden Partikeln, so müßte der damit verbundene Masseneintrag in die Fluidphase bzw. der Massenausstrom aus der Fluidphase in Form von Quelltermen berücksichtigt werden.

Formuliert man für das Kontrollvolumen Ω die Impulsbilanz, so erhält man:

$$\int_S \rho_F \vec{v}_F \vec{v}_F \cdot \vec{n} \, dS = \int_S \overline{\vec{T}} \cdot \vec{n} \, dS + \int_\Omega \rho_F \vec{g} \, d\Omega + \int_\Omega \vec{Q}_{P\vec{v}} \, d\Omega . \quad (2.3)$$

Die linke Seite dieser Gleichung stellt den Impulsfluß durch die Oberfläche von Ω dar. Die ersten beiden Terme auf der rechten Seite repräsentieren die äußeren Kräfte, die auf das in Ω enthaltene Fluid einwirken. Hierin ist $\overline{\overline{T}}$ der Spannungstensor, der sich für inkompressible Newtonsche Fluide in folgender Weise darstellen läßt:

$$\overline{\overline{T}} = -p \overline{\overline{I}} + 2\mu_F \overline{\overline{D}}, \quad (2.4)$$

wobei p der Druck, μ_F die dynamische Viskosität des Gases, $\overline{\overline{I}}$ der Einheitstensor und $\overline{\overline{D}}$ der Deformationstensor ist:

$$\overline{\overline{D}} = \frac{1}{2} \left[\text{grad } \vec{v}_F + (\text{grad } \vec{v}_F)^T \right]. \quad (2.5)$$

Die Größe \vec{g} in der Gleichung (2.3) ist die Gravitationsbeschleunigung. Andere äußere Kräfte, wie z.B. die an freien Oberflächen auftretende Oberflächenspannung oder elektromagnetische Kräfte, sollen hier nicht berücksichtigt werden. Der dritte Term auf der rechten Seite der Gleichung (2.3) repräsentiert Quellen bzw. Senken für den Impuls. Durch diesen Term wird der Einfluß der dispersen Phase auf die Bewegung der Fluidphase beschrieben, der durch unterschiedliche Geschwindigkeiten der beiden Phasen zustande kommt. Die Berechnung dieses Impulsquellterms wird in Kapitel 3.3.1 beschrieben.

Wie bereits oben erwähnt, sollen hier ausschließlich statistisch stationäre turbulente Strömungen betrachtet werden. Für die mathematische Beschreibung der Strömungsturbulenz existieren verschiedene Möglichkeiten. Für ingenieurtechnische Zwecke ist es in der Regel nicht notwendig, Details der turbulenten Schwankungsbewegungen in der Strömung zu kennen. Zur Auslegung eines Bauteils oder eines Apparates genügt es, integrale Größen wie den Druckverlust, die Kraft auf eine Fläche usw. zu berechnen. Diese Anforderungen werden für die meisten technisch relevanten Strömungen erfüllt, wenn man die Turbulenz mit Hilfe eines Turbulenzmodells beschreibt. In dieser Arbeit wurde ein k - ε -Modell verwendet, wobei k die turbulente kinetische Energie, d.h. die kinetische Energie der turbulenten Schwankungsbewegungen, und ε die Dissipationsrate der Turbulenzenergie ist. Für jede der beiden Größen wird eine Bilanzgleichung formuliert. Analog zu Kontinuitäts- und Impulsgleichung sollen diese Gleichungen hier in ihrer Integralform angegeben werden. Die Bilanzgleichung für die Turbulenzenergie lautet:

$$\int_S \rho_F k \vec{v}_F \cdot \vec{n} \, dS = \int_\Omega DF_k \, d\Omega + \int_\Omega P_k \, d\Omega - \int_\Omega DS_k \, d\Omega. \quad (2.6)$$

Hierin beschreibt der Ausdruck auf der linken Seite den konvektiven Transport von k , DF_k den diffusiven Transport, P_k die Produktion von Turbulenzenergie und DS_k die Vernichtung von Turbulenzenergie durch Dissipation. Die Bilanzgleichung für ε lautet:

$$\int_S \rho_F \varepsilon \vec{v}_F \cdot \vec{n} \, dS = \int_\Omega DF_\varepsilon \, d\Omega + \int_\Omega P_\varepsilon \, d\Omega - \int_\Omega DS_\varepsilon \, d\Omega, \quad (2.7)$$

wobei die linke Seite der Gleichung den konvektiven Transport von ε darstellt, DF_ε der diffusive Transport ist, P_ε die Produktion von ε und DS_ε die Dissipation von ε beschreibt. Die ausführlichen Berechnungsvorschriften für alle Terme der Gleichungen (2.6) und (2.7) werden im Abschnitt 2.1.2 angegeben.

2.1.2 Koordinatensystem

Die Gleichungen (2.2), (2.3), (2.6) und (2.7) bilden ein Gleichungssystem, das die Bewegung der fluiden Phase auf allgemeine Weise beschreibt. Zur Berechnung eines konkreten Strömungsproblems muß ein Koordinatensystem gewählt werden, auf dessen Basis die Komponenten der vektoriellen Größen der Bilanzgleichungen definiert werden. Hier werden die kartesischen Koordinaten (x, y) verwendet. Formuliert man für dieses Koordinatensystem die Kontinuitätsgleichung, so ergibt sich:

$$\int_S \rho_F (u_F n_x + v_F n_y) dS = 0, \quad (2.8)$$

worin u_F und v_F die kartesischen Komponenten des Geschwindigkeitsvektors und n_x und n_y die kartesischen Komponenten des Normalenvektors \vec{n} sind. Die Impulsbilanzgleichungen für die beiden Koordinatenrichtungen x und y ergeben sich aus der Gleichung (2.3) zu:

x -Impulsgleichung:

$$\begin{aligned} \int_S \rho_F u_F \vec{v}_F \cdot \vec{n} dS &= \int_S \mu_F \left[2 \frac{\partial u_F}{\partial x} n_x + \left(\frac{\partial u_F}{\partial y} + \frac{\partial v_F}{\partial x} \right) n_y \right] dS \\ &\quad - \int_\Omega \frac{\partial p}{\partial x} d\Omega + \int_\Omega \rho_F g_x d\Omega + \int_\Omega Q_{Pu} d\Omega, \end{aligned} \quad (2.9)$$

y -Impulsgleichung:

$$\begin{aligned} \int_S \rho_F v_F \vec{v}_F \cdot \vec{n} dS &= \int_S \mu_F \left[\left(\frac{\partial u_F}{\partial y} + \frac{\partial v_F}{\partial x} \right) n_x + 2 \frac{\partial v_F}{\partial y} n_y \right] dS \\ &\quad - \int_\Omega \frac{\partial p}{\partial y} d\Omega + \int_\Omega \rho_F g_y d\Omega + \int_\Omega Q_{Pv} d\Omega, \end{aligned} \quad (2.10)$$

mit

$$\vec{v}_F \cdot \vec{n} = u_F n_x + v_F n_y. \quad (2.11)$$

Die Indizes x und y kennzeichnen jeweils die entsprechende kartesische Komponente der indizierten vektoriellen Größe. Die turbulente kinetische Energie ergibt sich aus

der Summe der zeitlich gemittelten Quadrate der Schwankungsanteile der Geschwindigkeitskomponenten:

$$k = \frac{1}{2} \left(\overline{u_F'^2} + \overline{v_F'^2} + \overline{w_F'^2} \right) . \quad (2.12)$$

Für die einzelnen Terme der Bilanzgleichungen für die Turbulenzenergie und die Dissipationsrate ergibt sich in einem kartesischen Koordinatensystem folgende Darstellung:

$$DF_k = \frac{\partial}{\partial x} \left(\Gamma_k \frac{\partial k}{\partial x} \right) + \frac{\partial}{\partial y} \left(\Gamma_k \frac{\partial k}{\partial y} \right) , \quad (2.13)$$

$$P_k = \mu_t \left[2 \left[\left(\frac{\partial u_F}{\partial x} \right)^2 + \left(\frac{\partial v_F}{\partial y} \right)^2 \right] + \left(\frac{\partial u_F}{\partial y} + \frac{\partial v_F}{\partial x} \right)^2 \right] , \quad (2.14)$$

$$DS_k = \rho_F \varepsilon , \quad (2.15)$$

$$DF_\varepsilon = \frac{\partial}{\partial x} \left(\Gamma_\varepsilon \frac{\partial \varepsilon}{\partial x} \right) + \frac{\partial}{\partial y} \left(\Gamma_\varepsilon \frac{\partial \varepsilon}{\partial y} \right) , \quad (2.16)$$

$$P_\varepsilon = C_{\varepsilon 1} P_k \frac{\varepsilon}{k} , \quad (2.17)$$

$$DS_\varepsilon = \rho_F C_{\varepsilon 2} \frac{\varepsilon^2}{k} . \quad (2.18)$$

In diesen Gleichungen ist μ_t die scheinbare Viskosität (eddy viscosity):

$$\mu_t = \rho_F C_\mu \frac{k^2}{\varepsilon} , \quad (2.19)$$

die zusammen mit der molekularen Viskosität μ die Gesamtviskosität μ_F des turbulent strömenden Fluides ergibt:

$$\mu_F = \mu + \mu_t . \quad (2.20)$$

Die turbulente Viskosität ist in der Regel sehr viel größer als die laminare Viskosität. Die Größen Γ_k und Γ_ε sind Diffusionskoeffizienten, die sich wie folgt darstellen lassen:

$$\Gamma_k = \mu + \frac{\mu_t}{\sigma_k} , \quad \Gamma_\varepsilon = \frac{\mu_t}{\sigma_\varepsilon} . \quad (2.21)$$

In den oben angegebenen Gleichungen für das k - ε -Modell treten fünf Konstanten auf. Dabei handelt es sich um empirisch ermittelte Größen, für die hier die folgenden Standardwerte verwendet werden (siehe z.B. [36]):

$$C_\mu = 0.09 , \quad C_{\varepsilon 1} = 1.44 , \quad C_{\varepsilon 2} = 1.92 , \quad \sigma_k = 1.0 , \quad \sigma_\varepsilon = 1.3 . \quad (2.22)$$

Die Volumenintegrale über die Diffusionsterme DF_k und DF_ε lassen sich mit Hilfe des Gaußschen Integralsatzes in Oberflächenintegrale umwandeln:

$$\int_{\Omega} DF_k \, d\Omega = \int_S \Gamma_k \left(\frac{\partial k}{\partial x} n_x + \frac{\partial k}{\partial y} n_y \right) dS , \quad (2.23)$$

$$\int_{\Omega} DF_\varepsilon \, d\Omega = \int_S \Gamma_\varepsilon \left(\frac{\partial \varepsilon}{\partial x} n_x + \frac{\partial \varepsilon}{\partial y} n_y \right) dS . \quad (2.24)$$

Damit sind alle Gleichungen, die die Bewegung des Gases beschreiben, vollständig in kartesischen Koordinaten angegeben. Diese Gleichungen bilden die Grundlage für die numerische Berechnung der Gasströmung und sind der unmittelbare Ausgangspunkt für das im Abschnitt 2.2.2 beschriebene Diskretisierungsverfahren.

2.1.3 Randbedingungen

Im vorangehenden Abschnitt 2.1.2 wurde das Gleichungssystem angegeben, das das Verhalten der Strömung im Inneren eines Strömungsgebietes beschreibt. Zur vollständigen mathematischen Beschreibung und zur Berechnung eines konkreten Strömungsproblems ist es zusätzlich notwendig, die Werte Strömungsgrößen an den Rändern des Strömungsgebietes fest vorzugeben bzw. anzugeben, wie diese Werte aus anderen Größen berechnet werden können.

An Einströmrändern werden in dieser Arbeit Dirichletsche Randbedingungen verwendet, d.h. die Werte aller Strömungsgrößen werden fest vorgegeben. An Ausströmrändern wird für alle Variablen die von Neumannsche Randbedingung verwendet, d.h. der Gradient der Variablen normal zum Rand ist gleich Null.

Für Symmetrieränder gilt:

$$v_{Fn} = 0 , \quad \left(\frac{\partial \phi}{\partial n} \right)_{sym} = 0 , \quad \phi = v_{Ft} , p , k , \varepsilon , \quad (2.25)$$

wobei v_{Fn} und v_{Ft} die Komponenten des Geschwindigkeitsvektors normal bzw. tangential zum Symmetrierand sind und n die Koordinatenrichtung normal zum Rand bezeichnet. Der Normalgradient von v_{Fn} ist nicht zwingend gleich Null, was aus der Kontinuitätsgleichung folgt.

An festen, undurchlässigen Wänden gilt die Haftbedingung, d.h. die Geschwindigkeit des Fluides an der Wand ist gleich der Wandgeschwindigkeit, welche bei nicht bewegten Wänden gleich Null ist:

$$u_{Fw} = v_{Fw} = 0 . \quad (2.26)$$

Die Turbulenzgrößen an einer Wand ergeben sich zu:

$$k_w = 0, \quad \left(\frac{\partial \varepsilon}{\partial n} \right)_w = 0. \quad (2.27)$$

Zur Berechnung von ε in Wandnähe wird hier jedoch der folgende Ausdruck verwendet:

$$\varepsilon(h) = \frac{C_\mu^{3/4} [k(h)]^{3/2}}{\kappa h}. \quad (2.28)$$

Hierin ist κ die von Karmansche Konstante mit $\kappa = 0.4$ und h der Abstand zur Wand. Mit Hilfe dieser Beziehung wird die Dissipationsrate für den wandnächsten Punkt des numerischen Gitters berechnet. Die Voraussetzung für die Anwendbarkeit dieser Beziehung ist, daß dieser Punkt in der logarithmischen Region der turbulenten Grenzschicht liegt. Eine ausführliche Herleitung der Beziehung (2.28) ist z.B. in [36] zu finden.

2.2 Numerische Lösung der Bilanzgleichungen

Nachdem in den vorherigen Abschnitten das mathematische Modell angegeben wurde, das die Bewegung der fluiden Phase beschreibt, soll nun das in dieser Arbeit verwendete numerische Verfahren vorgestellt werden, mit dessen Hilfe ein konkretes Strömungsproblem gelöst wird.

In einem numerischen Verfahren werden die Strömungsgrößen stets an diskreten Punkten im Strömungsgebiet berechnet. Zu diesem Zweck muß das Strömungsgebiet in geeigneter Weise durch ein Netz diskreter Punkte beschrieben werden. Der Typ der hier verwendeten Gitternetze wird im Abschnitt 2.2.1 dargestellt. Bei der Diskretisierung des Strömungsgebietes durch ein Gitternetz entstehen Gitterzellen, die als Bilanzvolumen für die Volumen- und Oberflächenintegrale der Bilanzgleichungen dienen. Auf dieser Grundlage werden die Bilanzgleichungen diskretisiert, was im Abschnitt 2.2.2 beschrieben wird. Die diskretisierten Gleichungen werden linearisiert, und das daraus resultierende lineare Gleichungssystem wird mit einem iterativen Algorithmus gelöst, der im Abschnitt 2.2.3 erläutert wird. Zum Abschluß wird noch das Verfahren zur Berechnung des Drucks angegeben, der gesondert behandelt werden muß, da für ihn keine eigene Bilanzgleichung existiert.

2.2.1 Gitter für die numerische Simulation

In dieser Arbeit werden randangepaßte, nichtorthogonale, blockstrukturierte numerische Gitternetze verwendet. Diese Gitter bieten die Möglichkeit, auch komplexe

Strömungsgeometrien mit gekrümmten Berandungen, wie sie in der Verfahrenstechnik häufig vorhanden sind (z.B. Rohrleitungssysteme, Düsen, usw.), zu vernetzen. Die Gitterlinien folgen der Berandung des Strömungsgebietes, was die Implementierung von Randbedingungen vereinfacht. Des weiteren existiert die Möglichkeit, in Gebieten, in denen sich die Strömungsgrößen stark ändern, ein feineres Gitter zu verwenden und so die Genauigkeit der numerisch berechneten Lösung zu erhöhen. Ein Nachteil nichtorthogonaler Gitter gegenüber orthogonalen Gittern besteht darin, daß die diskretisierten Bilanzgleichungen zusätzliche Terme enthalten und sich dadurch der Rechenaufwand erhöht. Dieser Nachteil wird aber hier in Kauf genommen, weil er durch den Vorteil der geometrischen Flexibilität aufgewogen wird. Die Anordnung der Strömungsvariablen im Gitternetz ist nicht-versetzt ("colocated grid"), d.h. alle Variablen werden für den Mittelpunkt der Gitterzelle gespeichert.

2.2.2 Diskretisierung der Bilanzgleichungen nach der Finite-Volumen-Methode

Unter Berücksichtigung der im Abschnitt 2.2.1 beschriebenen Geometrie des numerischen Gitters soll im folgenden die Diskretisierung der Bilanzgleichungen beschrieben werden. Hierzu wird die Finite-Volumen-Methode verwendet. Die in den Bilanzgleichungen auftretenden Oberflächen- und Volumenintegrale müssen durch Berechnungsvorschriften approximiert werden, in denen allein die Werte der Strömungsvariablen in den Zellenmittelpunkten verwendet werden. In der Abbildung 2.2 werden die Bezeichnungen erläutert, die in diesem Abschnitt verwendet werden.

Berechnung der Volumenintegrale

Das Volumenintegral einer Größe q wird aus dem Produkt des Mittelwerts \bar{q} dieser Größe bezogen auf die Gitterzelle und dem Volumen der Gitterzelle berechnet. Dabei wird \bar{q} durch den Wert von q im Zellenmittelpunkt approximiert:

$$\int_{\Omega} q \, d\Omega = \bar{q} \, \Delta\Omega \approx q_P \, \Delta\Omega . \quad (2.29)$$

Falls q eine zusammengesetzte Größe ist, kann der Wert von q_P auf einfache Weise berechnet werden, da die Werte aller Variablen im Zellenmittelpunkt bekannt sind.

Berechnung der Oberflächenintegrale

Das Integral einer Größe f über die Oberfläche einer Gitterzelle wird aus der Summe der Integrale über die einzelnen Seitenflächen gebildet:

$$F_e = \int_S f \, dS = \sum_{k=1}^4 \int_{S_k} f \, dS . \quad (2.30)$$

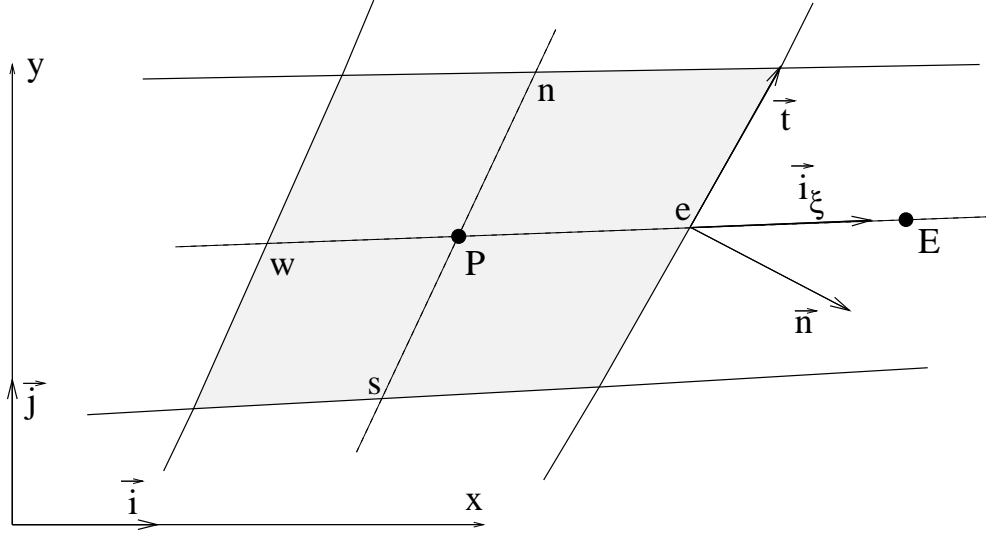


Abbildung 2.2: Bezeichnungen für eine Gitterzelle

Da alle Seitenflächen analog behandelt werden, seien die folgenden Ausführungen auf die mit e (= east) bezeichnete “östliche“ Seitenfläche beschränkt. Das Integral von f über eine Seitenfläche ergibt sich aus dem Produkt des Mittelwertes von f bezogen auf diese Fläche und dem Flächeninhalt:

$$\int_{S_e} f \, dS = (\bar{f})_e S_e \approx f_e S_e , \quad (2.31)$$

wobei S_e der Flächeninhalt und f_e der Wert von f im Mittelpunkt der Fläche ist.

Berechnung der konvektiven Flüsse

Auf der linken Seite der im Abschnitt 2.1.2 angegebenen Bilanzgleichungen steht jeweils ein Ausdruck, der einen konvektiven Fluß repräsentiert. Er hat die allgemeine Form:

$$F^c = \int_S \rho_F \phi \vec{v}_F \cdot \vec{n} \, dS , \quad (2.32)$$

wobei ϕ die Variable ist, deren Bilanz von der jeweiligen Gleichung beschrieben wird. Für die Kontinuitätsgleichung ist $\phi = 1$.

Der Massenstrom durch die Seitenfläche e einer Gitterzelle läßt sich näherungsweise berechnen aus:

$$\dot{m}_e = \int_{S_e} \rho_F \vec{v}_F \cdot \vec{n} \, dS \approx (\rho_F \vec{v}_F \cdot \vec{n})_e S_e = \rho_F (u_F S_x + v_F S_y)_e , \quad (2.33)$$

wobei $\vec{S}_e = \vec{n}_e S_e$ der Flächenvektor der Seitenfläche ist. Nach Einsetzen in die Gleichung (2.32) ergibt sich die Approximation des konvektiven Flusses durch die Seitenfläche zu:

$$F_e^c \approx \dot{m}_e \phi_e . \quad (2.34)$$

Die Berechnung dieses Ausdrucks wird mit Hilfe der Deferred–Correction Methode nach Khosla und Rubin [23] durchgeführt:

$$F_e^c \approx (F_e^{cL})^{neu} + \beta (F_e^{cH} - F_e^{cL})^{alt} . \quad (2.35)$$

Hierin kennzeichnet das Superskript neu einen implizit formulierten Ausdruck, der die noch unbekannten Variablenwerte der nächsten Nachbarpunkte für den aktuellen Iterationsschritt enthält, und die mit dem Superskript alt gekennzeichneten Terme werden explizit mit bereits bekannten Variablenwerten aus der vorhergehenden Iteration berechnet. Das Superskript L (= Lower order) bezeichnet Terme, deren Größe im Mittelpunkt der Seitenfläche mit Hilfe der Upwind–Interpolation:

$$\phi_e = \begin{cases} \phi_P & \text{falls } (\vec{v}_F \cdot \vec{n})_e > 0 \\ \phi_E & \text{falls } (\vec{v}_F \cdot \vec{n})_e < 0 \end{cases} \quad (2.36)$$

berechnet werden, wobei ϕ_P und ϕ_E die Werte der Größe ϕ in den Punkten P bzw. E sind (siehe Abbildung 2.2). Der mit dem Superskript H (= Higher order) bezeichnete Term wird mit Hilfe einer linearen Interpolation berechnet:

$$\phi_e = \phi_E \lambda_e + \phi_P (1 - \lambda_e) , \quad \text{mit} \quad \lambda_e = \frac{|\vec{r}_e - \vec{r}_P|}{|\vec{r}_E - \vec{r}_P|} . \quad (2.37)$$

Der Wichtungsfaktor β in der Gleichung 2.35 wurde hier mit $\beta = 0.1$ gewählt.

Berechnung der diffusiven Flüsse

Der erste Term auf der rechten Seite der Impulsbilanzgleichungen bzw. der Bilanzgleichungen für k und ε hat die allgemeine Form:

$$\underbrace{\int_S \Gamma \text{grad } \phi \cdot \vec{n} \, dS}_{F^d} + A , \quad (2.38)$$

mit

$$\phi = u_F, v_F, k, \varepsilon , \quad \text{und} \quad A \equiv 0 \quad \text{für } \phi = k, \varepsilon . \quad (2.39)$$

Hierin repräsentiert F^d einen diffusiven Fluß, wobei Γ der zugehörige Diffusionskoeffizient ist. Der diffusive Fluß durch die Seitenfläche e wird nach Perić [36] approximiert durch:

$$F_e^d \approx \Gamma_e \frac{S_e}{L_{P,E}} (\phi_E - \phi_P)^{neu} + \Gamma_e S_e (\text{grad } \phi)_e^{alt} \cdot (\vec{n} - \vec{i}_\xi) . \quad (2.40)$$

Hierbei ist $L_{P,E}$ der Abstand der Punkte P und E , und die Superskripte ^{neu} bzw. ^{alt} bezeichnen wiederum implizit bzw. explizit berechnete Terme. Eine kurze Herleitung dieser Approximation ist im Anhang A.1 angegeben. Eine ausführliche Herleitung findet sich z.B. in [36].

Der in den Impulsgleichungen vorkommende Term A mit:

$$A_{x_i} = \int_S \Gamma \left(\frac{\partial u_F}{\partial x_i}, \frac{\partial v_F}{\partial x_i} \right)^T \cdot \vec{n} \, dS, \quad x_i = x, y \quad (2.41)$$

wird auf einfache Weise explizit berechnet. Zum Beispiel wird für die x -Impulsgleichung die folgende Approximation verwendet:

$$A_{x_e} \approx \Gamma_e S_e \left[\left(\frac{\partial u_F}{\partial x} \right)_e^{alt} + \left(\frac{\partial v_F}{\partial x} \right)_e^{alt} \right], \quad (2.42)$$

Die in den Gleichungen (2.40) und (2.42) benötigten Ableitungen der Geschwindigkeitskomponenten im Mittelpunkt der Seitenfläche werden durch lineare Interpolation der Ableitungen in den benachbarten Zellenmittelpunkten bestimmt. Die Ableitung im Zellenmittelpunkt wird mit Hilfe der Beziehung:

$$\left(\frac{\partial \phi}{\partial x} \right)_P \approx \frac{\sum_c \phi_c S_{cx}}{\Delta \Omega}, \quad c = e, n, w, s \quad (2.43)$$

berechnet [36], für die im Anhang A.2 eine kurze Herleitung angegeben ist.

2.2.3 Lösungsmethode

Nach der Durchführung der im Abschnitt 2.2.2 beschriebenen Diskretisierung liegt für jede der unbekannten Variablen u_F , v_F , k und ε ein Gleichungssystem vor, in dem die Anzahl der Gleichungen der Anzahl der Gitterzellen des numerischen Gitters entspricht. Diese Gleichungssysteme sind miteinander gekoppelt. Die Lösung der einzelnen Gleichungssysteme sowie des gekoppelten Gesamtsystems erfolgt auf iterativem Wege. Dabei wird zunächst für jede Variable das zugehörige Gleichungssystem unabhängig von den übrigen Variablen iterativ gelöst (innere Iteration). Dies geschieht mit Hilfe der SIP-Methode nach Stone [44], die im Anhang A.3 zusammengefaßt dargestellt ist. Die Werte der anderen Variablen, die in diesem Gleichungssystem vorkommen, werden als bekannt angenommen. Mit den neu berechneten Variablenwerten werden die Koeffizientenmatrizen und Quelltermvektoren der einzelnen Gleichungssysteme neu bestimmt und der gesamte Vorgang wiederholt (äußere Iteration).

Bei der Neuberechnung des Geschwindigkeitsfeldes für einen äußeren Iterationsschritt wird ein Druckfeld verwendet, das aus dem vorherigen Iterationsschritt stammt. Da der Druck und die Geschwindigkeit über die Impulsgleichungen stark gekoppelt sind, hat dies zur Folge, daß das neue Geschwindigkeitsfeld nicht die Kontinuitätsgleichung erfüllt. Zur Bestimmung des korrekten Geschwindigkeitsfeldes muß zunächst der Druck neu berechnet werden. Mit Hilfe des neuen Druckfeldes können dann die Geschwindigkeitskomponenten korrigiert werden, so daß die Kontinuitätsgleichung erfüllt wird.

Zur Berechnung des Drucks und der Korrektur der Geschwindigkeitswerte wird hier das SIMPLE-Verfahren nach Caretto [5] bzw. Patankar [35] verwendet, das von Peric [36] für eine nicht-versetzte Anordnung der Variablen im numerischen Gitter angepaßt wurde.

Kapitel 3

Berechnung der Bewegung der Partikelphase

In diesem Kapitel sollen die Komponenten des mathematischen Modells beschrieben werden, mit dessen Hilfe die Bewegung der dispersen Phase berechnet wird. Die Partikel werden dabei als fest und kugelförmig angenommen, und die zu einem bestimmten Zeitpunkt in einer Strömung vorhandenen Partikel können unterschiedliche Durchmesser haben.

Im Abschnitt 3.1 werden zunächst einige wichtige Parameter und Begriffe definiert, die der Klassifizierung von Gas-Partikel-Strömungen dienen und im folgenden häufig verwendet werden. Eine Möglichkeit zur Berechnung von Gas-Partikel-Strömungen bietet das Zwei-Fluid-Modell, in dem die beiden Phasen als sich gegenseitig durchdringende Fluide angesehen werden. Eine kurze Beschreibung dieses Modells erfolgt im Abschnitt 3.2. In dieser Arbeit wird zur Beschreibung der Bewegung der Partikelphase jedoch das Lagrange-Modell verwendet, das die Bewegung diskreter Partikel in dem umgebenden Fluid modelliert. Im Abschnitt 3.3 wird dieses Modell ausführlich erläutert. Die Modelle für die Partikel-Partikel-Kollisionen werden in diesem Kapitel nicht behandelt. Sie werden separat im Kapitel 4 beschrieben.

3.1 Parameter zur Klassifizierung von Gas-Partikel-Strömungen

Gas-Partikel-Strömungen besitzen eine Reihe charakteristischer Merkmale, anhand derer eine Klassifizierung möglich ist. Ein wichtiger Parameter ist die Massenbelastung η , die als das Verhältnis der Massenströme von disperser und fluider Phase

definiert ist:

$$\eta = \frac{\dot{m}_P}{\dot{m}_F} . \quad (3.1)$$

Ein weiterer wichtiger Parameter ist der relative Volumenanteil α , der die Größe des Volumens angibt, das eine Phase pro Einheitsvolumen des Gemisches einnimmt. Hierbei ist α_P der Volumenanteil der Partikel und $\alpha_F = 1 - \alpha_P$ der Volumenanteil des Fluides. Die Gemischdichte σ einer Phase ist die Masse dieser Phase pro Volumeneinheit des Phasengemisches. Die Gemischdichte für die Partikelphase läßt sich aus dem Volumenanteil und der Materialdichte der Partikel bestimmen zu:

$$\sigma_P = \alpha_P \rho_P . \quad (3.2)$$

Analog dazu ergibt sich die Gemischdichte des Fluides zu:

$$\sigma_F = (1 - \alpha_P) \rho_F , \quad (3.3)$$

wobei ρ_F die Fluidichte ist. Zur Gemischdichte seien im folgenden noch einige Anmerkungen gemacht. In einem Kontinuum ist die Dichte als das Verhältnis von Masse zu einem Einheitsvolumen definiert. Dieses Einheitsvolumen ist infinitesimal klein, d.h. die Dichte ist für einen Punkt definiert. Zur Ermittlung der Dichte in einem realen Gas muß ein Volumen endlicher Größe betrachtet werden, das so groß sein muß, daß es eine ausreichende Anzahl von Molekülen für eine statistisch genaue Mittelwertbildung enthält. Um die Mittelwertbildung mit einem Fehler von weniger als 1% durchführen zu können, muß das Kontrollvolumen etwa 10^4 Moleküle enthalten [37]. Bei einem Gas unter Normalbedingungen ist hierzu z.B. ein würfelförmiges Volumen mit einer Kantenlänge von ca. $0.1 \mu\text{m}$ erforderlich [9]. In fast allen strömungsmechanischen Anwendungen ist dieses Volumen um mehrere Größenordnungen kleiner als die charakteristische Abmessung des Strömungsgebietes. Es kann deshalb in diesen Fällen als punktförmig angesehen werden.

Um für die disperse Phase eine Gemischdichte mit hinreichender Genauigkeit bestimmen zu können, muß ebenfalls ein Kontrollvolumen betrachtet werden, das ausreichend viele Partikel enthält. Betrachtet man z.B. ein Gemisch von Kohlepartikeln mit einem Durchmesser von $100 \mu\text{m}$ und Luft bei einer Massenbeladung von $\eta = 1$, so müßte ein würfelförmiges Kontrollvolumen, das wiederum ca. 10^4 Kohlepartikel enthält, eine Kantenlänge von ca. 1 cm haben! Bei vielen Strömungen in der Verfahrens- bzw. Energietechnik liegen die charakteristischen Strömungsdimensionen in einer ähnlichen Größenordnung. In diesen Fällen kann eine Gemischdichte für die disperse Phase also nicht sinnvoll bestimmt werden, und die Partikelströmung kann deshalb nicht als Kontinuumsströmung angesehen werden. Die Anwendung von Erhaltungsgleichungen für integrale Größen zur Beschreibung der Partikelbewegung, wie sie z.B. in dem im Abschnitt 3.2 beschriebenen Zwei-Fluid-Modell erfolgt, ist in solchen Fällen nicht möglich.

Ein weiteres wichtiges Merkmal zur Charakterisierung einer Gas–Partikel–Strömung ist die Angabe, ob es sich um eine sogenannte dichte oder eine verdünnte Strömung handelt. Eine Gas–Partikel–Strömung wird als verdünnt bezeichnet, wenn die Bewegung eines Partikels, abgesehen von Körperkräften wie z.B. der Gravitationskraft, im wesentlichen durch die aerodynamischen Kräfte bestimmt wird, die vom Gas auf das Partikel ausgeübt werden. Im Gegensatz dazu wird eine Strömung, in der die Partikelbewegung überwiegend durch die Kollisionen zwischen den Partikeln beeinflusst wird, als dicht bezeichnet. Ein Parameter zur Unterscheidung zwischen verdünnten und dichten Strömungen ist das Verhältnis der aerodynamischen Relaxationszeit τ_A zur mittleren Zeit zwischen zwei Kollisionen τ_C . Die aerodynamische Relaxationszeit ist ein Zeitmaß, das angibt, wie schnell sich die Geschwindigkeit eines Partikels ändert, wenn sich die Geschwindigkeit des umgebenden Fluides ändert. Diese Zeit läßt sich näherungsweise bestimmen durch [8]:

$$\tau_A = \frac{\rho_P d_P^2}{18 \mu}, \quad (3.4)$$

wobei d_P der Partikeldurchmesser und μ die molekulare Viskosität des Gases ist. Im Fall einer verdünnten Strömung hat das Partikel zwischen zwei Kollisionen relativ viel Zeit, um auf die aerodynamischen Kräfte des Fluides zu reagieren. Deshalb wird in diesem Fall die Partikelbewegung überwiegend durch die Wechselwirkung mit dem Fluid bestimmt. Für den anderen Grenzfall, d.h. einer dichten Strömung, ist die Ansprechzeit des Partikels auf die Fluidkräfte wesentlich größer als die Zeit bis zur nächsten Kollision. Die Partikelbewegung wird dann durch die Partikel–Partikel–Kollisionen dominiert.

Ein weiterer wichtiger Klassifizierungsparameter ist die Stokes–Zahl. Sie ist definiert als das Verhältnis der aerodynamischen Relaxationszeit des Partikels und einer charakteristischen Zeit τ_S des Strömungssystems:

$$\text{St} = \frac{\tau_A}{\tau_S}. \quad (3.5)$$

Ist die Stokes–Zahl sehr klein, so hat das Partikel ausreichend Zeit, um auf Geschwindigkeitsänderungen des umgebenden Gases zu reagieren. Ein solches Partikel folgt der Fluidströmung nahezu ideal, d.h. in einer stationären Strömung sind die Bahnlinien solcher Partikel nahezu identisch mit den Stromlinien des Fluides. Ist dagegen die Stokes–Zahl sehr groß, so reagiert das Partikel nur sehr langsam auf Änderungen der Fluidströmung. Die Bewegung eines solchen Partikels wird stark durch dessen Massenträgheit bestimmt, was in der Regel häufige Kollisionen des Partikels mit der Strömungsberandung zur Folge hat.

Als letzter Aspekt zur Charakterisierung von Gas–Partikel–Strömungen sei hier die Kopplung zwischen der dispersen und der fluiden Phase betrachtet. Bei geringen Beladungen hat das Vorhandensein der Partikel keine signifikanten Auswirkungen auf

das Geschwindigkeitsfeld des Gases. In diesem Fall spricht man von einer Ein-Weg-Kopplung. Liegt eine mittlere bis hohe Beladung vor, so hat die Partikelbewegung eine merkliche Rückwirkung auf die Fluidbewegung. Im Falle einer solchen Zwei-Wege-Kopplung findet eine Wechselwirkung zwischen den Phasen statt, die auf einer gegenseitigen Impulsübertragung beruht. Weitere Arten der Phasenwechselwirkung sind z.B. Massen- oder Wärmeübertragung, die bei verbrennenden, verdampfenden oder kondensierenden Partikeln eine wichtige Rolle spielen. Bei den in dieser Arbeit betrachteten Gasströmungen mit dispergierten festen Partikeln ist jedoch nur die Wechselwirkung durch Impulsübertragung vorhanden.

3.2 Das Euler/Euler– oder Zwei-Fluid-Modell

Eine Möglichkeit zur mathematischen Beschreibung einer dispersen Zweiphasenströmung ist das Zwei-Fluid-Modell. In diesem Modell wird angenommen, daß die beiden Phasen zwei kontinuierliche Fluide sind, die sich gegenseitig durchdringen und miteinander in Wechselwirkung stehen. Die Gleichungen zur Beschreibung dieser Strömung erhält man, indem die Massen- und Impulserhaltungsgleichungen für beide Fluide über ein Kontrollvolumen gemittelt werden. Für eine zweidimensionale stationäre Strömung und ohne die Berücksichtigung einer Massenübertragung zwischen den Phasen lautet die Kontinuitätsgleichung für die disperse Phase:

$$\frac{\partial(\sigma_P u_P)}{\partial x} + \frac{\partial(\sigma_P v_P)}{\partial y} = 0 , \quad (3.6)$$

wobei u_P und v_P die kartesischen Geschwindigkeitskomponenten der dispersen Phase sind. Analog dazu lautet die Kontinuitätsgleichung für das Fluid:

$$\frac{\partial(\sigma_F u_F)}{\partial x} + \frac{\partial(\sigma_F v_F)}{\partial y} = 0 . \quad (3.7)$$

Die Impulsgleichungen für die beiden Phasen ergeben sich im Zwei-Fluid-Modell zu:

$$\begin{aligned} \sigma_P u_P \frac{\partial u_P}{\partial x} + \sigma_P v_P \frac{\partial u_P}{\partial y} &= f_{PFx} , \\ \sigma_P u_P \frac{\partial v_P}{\partial x} + \sigma_P v_P \frac{\partial v_P}{\partial y} &= f_{PFy} , \end{aligned} \quad (3.8)$$

für die Partikelphase sowie

$$\begin{aligned} \sigma_F u_F \frac{\partial u_F}{\partial x} + \sigma_F v_F \frac{\partial u_F}{\partial y} &= -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} - f_{PFx} , \\ \sigma_F u_F \frac{\partial v_F}{\partial x} + \sigma_F v_F \frac{\partial v_F}{\partial y} &= -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} - f_{PFy} \end{aligned} \quad (3.9)$$

für die Fluidphase. Hierin sind die τ_{ij} die Komponenten des Spannungstensors, und \vec{f}_{PF} ist die von den Partikeln pro Volumeneinheit auf das Fluid ausgeübte Kraft.

Der Vorteil des Zwei-Fluid-Modells besteht darin, daß die Gleichungen, die die Bewegung der dispersen Phase beschreiben, dieselbe Form haben wie die Gleichungen für die Fluidbewegung. Zur Lösung dieser Gleichungen können daher nahezu dieselben Algorithmen verwendet werden, wie sie auch bei der Simulation der Fluidströmung zum Einsatz kommen. Diesem Vorteil stehen eine Reihe von Nachteilen gegenüber. Neben den grundsätzlichen Problemen (siehe Abschnitt 3.1) ist es in der Regel schwierig, allgemeingültige Ansätze für die Formulierung der Schubspannung der dispersen Phase, für die Partikeldispersion oder auch für die Diffusion kinetischer Energie zu finden [10]. Hierzu werden meistens Gradienten-Diffusions-Ansätze verwendet. Diese beinhalten eine Reihe empirischer Konstanten, deren optimale Werte aus dem Vergleich von Simulationen und Experimenten ermittelt werden müssen [14]. Eine weitere Schwierigkeit besteht in der Formulierung geeigneter Randbedingungen, so z.B. für die Wechselwirkung der dispersen Phase mit festen Wänden. Des weiteren steigt der Rechenzeit- und Speicheraufwand einer Simulation, wenn die disperse Phase polydispers ist, d.h. aus Partikeln unterschiedlichen Durchmessers besteht. In diesem Fall werden alle Partikel eines bestimmten Durchmessers als separate Phase betrachtet.

Die Anwendung des Zwei-Fluid-Modells ist nur dann möglich, wenn in der untersuchten Strömung die Anzahldichte der Partikel n_P so groß ist, daß die Berechnung mittlerer lokaler Größen für die disperse Phase sinnvoll durchgeführt werden kann. Die Bedingungen, unter denen dies möglich ist, wurden bereits im Abschnitt 3.1 erläutert.

3.3 Das Euler/Lagrange-Verfahren

In dieser Arbeit wird zur Berechnung der Bewegung der dispersen Phase das Lagrange-Modell verwendet. Die disperse Phase wird dabei als eine Menge diskreter Einzelpartikel angesehen, deren Lagrangesche Bewegungsgleichungen numerisch gelöst werden. Um den Rechenaufwand zu begrenzen, werden in der Simulation im allgemeinen weniger Partikel betrachtet, als in der realen Strömung vorhanden sind. Jedes simulierte Partikel repräsentiert daher eine bestimmte Anzahl physikalischer Partikel mit denselben physikalischen Eigenschaften. Die Bewegung eines einzelnen Partikels resultiert aus den an diesem Partikel angreifenden Kräften und Momenten. Die Berechnung der Bewegung eines Einzelpartikels nach dem Lagrange-Verfahren wird im Abschnitt 3.3.1 beschrieben. Aus der Summe der einzelnen Partikelbewegungen ergeben sich die makroskopischen Strömungsgrößen der dispersen Phase,

wie z.B. die lokale Partikelkonzentration oder die mittlere Partikelgeschwindigkeit. Zur Bestimmung dieser Größen existieren zwei prinzipiell verschiedene Verfahren, nämlich zum einen die sukzessive Berechnung einer großen Anzahl von Trajektorien, und zum anderen die simultane Berechnung der Bewegung aller simulierten Partikel. Diese beiden Verfahren werden im Abschnitt 3.3.2 erläutert.

3.3.1 Berechnung der Bewegung eines Einzelpartikels

Gleichungen für die translatorische Partikelbewegung

Ein Partikel, das sich in einem Strömungsgebiet bewegt, ändert seinen momentanen Bewegungszustand, wenn äußere Kräfte auf es einwirken. Diese Kräfte können z.B. aerodynamische Kräfte sein, die das Fluid auf das Partikel ausübt. Des weiteren treten Körperkräfte wie die Gravitationskraft oder gegebenenfalls elektrische bzw. magnetische Kräfte auf. Auch Stoßkräfte, die bei der Kollision des Partikels mit einer Wand oder mit einem anderen Partikel auftreten, haben eine Änderung des Bewegungszustands des Partikels zur Folge. In diesem Abschnitt soll zunächst nur der Einfluß der aerodynamischen Kräfte und der Körperkräfte auf die Partikelbewegung betrachtet werden.

Die translatorische Bewegung des Partikels wird durch die Impulsgleichung beschrieben, wonach die Änderung des Partikelimpulses gleich der Summe der auf das Partikel einwirkenden äußeren Kräfte ist. Für ein Partikel mit der Masse m_P lautet die Impulsgleichung:

$$m_P \frac{d\vec{v}_P}{dt} = \vec{F}_W + \vec{F}_M + \vec{F}_S + \vec{F}_G . \quad (3.10)$$

Hierbei ist \vec{F}_W die Widerstandskraft, \vec{F}_M die Querkraft aufgrund der Partikelrotation (Magnus-Kraft), \vec{F}_S die Querkraft durch eine Scheranströmung des Partikels (Saffman-Kraft) und \vec{F}_G die Gravitationskraft.

Die aerodynamische Widerstandskraft \vec{F}_W auf ein Partikel in einer parallelen Fluidströmung resultiert aus einem Reibungs- und einem Druckwiderstand. Der Reibungswiderstand wird durch die Haftbedingung an der Partikeloberfläche verursacht. Der Druckwiderstand resultiert aus einer ungleichmäßigen Druckverteilung an der Partikeloberfläche. Dabei ist der Druck, der auf die stromab gelegene Seite des Partikels wirkt, kleiner als der auf die stromauf gelegene Seite wirkende Druck. Die resultierende Druckkraft ist der Partikelbewegung entgegengerichtet.

Die Gleichung zur Bestimmung der Widerstandskraft lautet:

$$\vec{F}_W = \frac{\pi}{8} \rho_F d_P^2 c_W v_{rel} \vec{v}_{rel} , \quad (3.11)$$

wobei \vec{v}_{rel} die Relativgeschwindigkeit zwischen Fluid und Partikel ist:

$$\vec{v}_{rel} = \vec{v}_F - \vec{v}_P , \quad (3.12)$$

und

$$v_{rel} = |\vec{v}_{rel}| = \sqrt{(u_F - u_P)^2 + (v_F - v_P)^2} . \quad (3.13)$$

Der zur Bestimmung von \vec{F}_W benötigte Widerstandsbeiwert c_W ist eine meistens empirische bestimmte Größe, die von der Partikel–Reynoldszahl abhängig ist:

$$Re_P = \frac{d_P v_{rel}}{\nu_F} . \quad (3.14)$$

Hierin ist ν_F die kinematische Viskosität des Fluides. Zur Bestimmung des Widerstandsbeiwertes wurden in der Vergangenheit umfangreiche Untersuchungen durchgeführt (siehe z.B. [6, 33]). In dieser Arbeit wurde die folgende von Morsi und Alexander [33] angegebene Formel zur Berechnung von c_W in Abhängigkeit von Re_P verwendet:

$$c_W = a Re_P^{-2} + b Re_P^{-1} + c . \quad (3.15)$$

Die Werte der Parameter a , b und c sind wiederum von der Größe der Partikel–Reynoldszahl abhängig, wie in der Tabelle 3.1 angegeben. In der Abbildung 3.1 ist die Abhängigkeit des Widerstandsbeiwertes von der Partikel–Reynoldszahl graphisch dargestellt. Werte von Re_P , die größer als die in der Tabelle 3.1 bzw. der Abbildung 3.1 angegebenen sind, haben für die hier betrachteten Gas–Partikel–Strömungen keine praktische Relevanz.

Die Magnus–Kraft \vec{F}_M ist eine Querkraft, die auf ein rotierendes Partikel in einer parallelen Fluidströmung wirkt. Aufgrund der an der Partikeloberfläche geltenden Haftbedingung verändert sich das Fluidgeschwindigkeitsfeld in der Nähe des rotierenden Partikels. Auf der Seite des Partikels, die sich in Richtung der Fluidströmung bewegt, wird das Fluid beschleunigt, und auf der gegenüberliegenden Seite wird es verzögert (siehe Abbildung 3.2). Dadurch sinkt auf der beschleunigten Seite der Druck, wogegen er auf der verzögerten Seite ansteigt. Aus dieser Druckdifferenz resultiert eine Querkraft auf das Partikel, die senkrecht zur Richtung der Fluidströmung gerichtet ist.

Die Gleichung zur Bestimmung der Magnus–Kraft lautet:

$$\vec{F}_M = \frac{\pi}{8} \rho_F d_P^2 c_M \frac{v_{rel}}{\omega_{rel}} (\vec{\omega}_{rel} \times \vec{v}_{rel}) . \quad (3.16)$$

Hierbei ist $\vec{\omega}_{rel}$ die Rotation des Partikels relativ zum Fluid:

$$\vec{\omega}_{rel} = \vec{\omega}_F - \vec{\omega}_P , \quad (3.17)$$

Re_P	a	b	c
$0 < Re_P < 0.1$	0	24	0
$0.1 \leq Re_P < 1$	0.0903	22.73	3.69
$1 \leq Re_P < 10$	-3.8889	29.1667	1.222
$10 \leq Re_P < 100$	-116.67	46.5	0.6167
$100 \leq Re_P < 1000$	-2778.0	98.33	0.3644
$1000 \leq Re_P < 5000$	-47500	148.62	0.357
$5000 \leq Re_P < 10000$	578700	-490.546	0.46
$10000 \leq Re_P < 50000$	5416700	-166.25	0.5191
$50000 \leq Re_P$	0	0	0.49

Tabelle 3.1: Werte für die Parameter a , b und c zur Berechnung des Widerstandsbeiwertes c_W

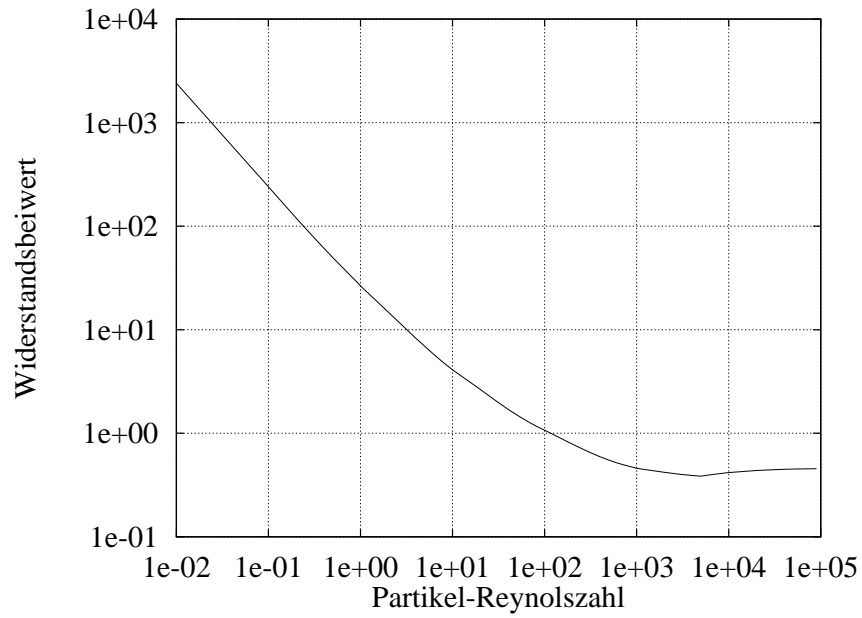


Abbildung 3.1: Größe des Widerstandsbeiwertes c_W in Abhängigkeit von der Partikel-Reynoldszahl Re_P

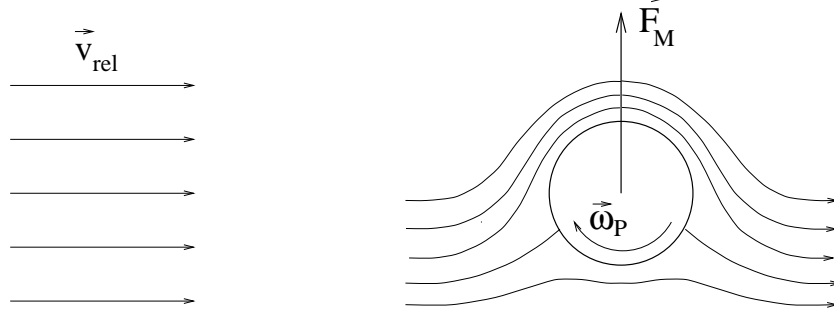


Abbildung 3.2: Querkraft auf ein rotierendes Partikel in einer Parallelströmung

mit

$$\omega_{rel} = |\vec{\omega}_{rel}| = \sqrt{(\omega_{Fx} - \omega_{Px})^2 + (\omega_{Fy} - \omega_{Py})^2} . \quad (3.18)$$

Die Rotation des Fluides am Partikelort ergibt sich aus:

$$\vec{\omega}_F = \nabla \times \vec{v}_F = \frac{\partial v_F}{\partial x} - \frac{\partial u_F}{\partial y} . \quad (3.19)$$

Zur Bestimmung des Beiwertes der Magnus-Kraft c_M wurden u.a. von Tsuji et al. [49] Untersuchungen durchgeführt. Daraus wurden die folgenden Beziehungen abgeleitet, die auch in dieser Arbeit verwendet wurden:

$$c_M = \begin{cases} (0.4 \pm 0.1) \sigma_M & \text{für } \sigma_M < 1 , \\ (0.4 \pm 0.1) & \text{für } \sigma_M \geq 1 , \end{cases} \quad (3.20)$$

mit

$$\sigma_M = \frac{1}{2} \frac{d_P \omega_{rel}}{v_{rel}} . \quad (3.21)$$

Wenn die Rotationsgeschwindigkeit hinreichend groß ist, kann die Magnus-Kraft die Bewegung eines Partikels signifikant beeinflussen. Solche hohen Rotationsgeschwindigkeiten sind vor allem für Gas-Partikel-Strömungen in Kanälen und Rohren zu beobachten. Hierbei kommt es zu häufigen Partikel-Wand-Kollisionen, bei denen durch die Reibung zwischen der Partikeloberfläche und der Wand relativ große Tangentialkräfte auf die Partikeloberfläche wirken. Beim pneumatischen Transport von Partikeln mit einem Durchmesser von 0.5–1 mm in einem rechteckigen Kanal wurden z.B. von Matsumoto und Saito [27] Rotationsgeschwindigkeiten von 300–2000 rad/s gemessen.

Die Saffman-Kraft \vec{F}_S ist eine Querkraft, die entsteht, wenn sich ein Partikel in einer Scherströmung bewegt. Betrachtet man ein ruhendes Partikel, so bedeutet dies, daß

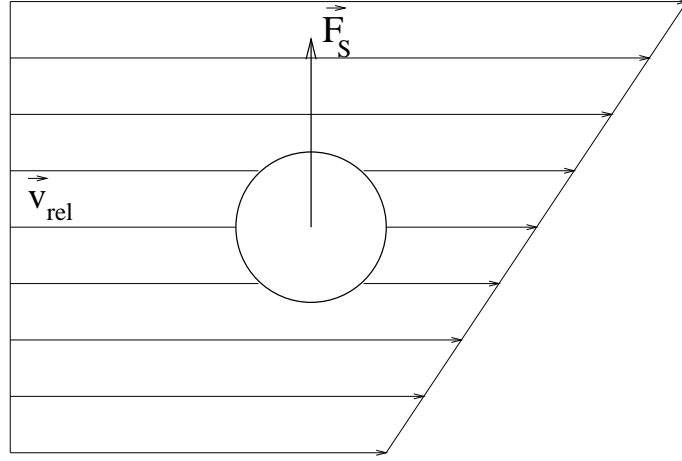


Abbildung 3.3: Querkraft auf ein Partikel in einer Scherströmung

das Geschwindigkeitsfeld des Fluides am Partikelort einen Gradienten senkrecht zur Richtung der mittleren Anströmgeschwindigkeit aufweist (siehe Abbildung 3.3). Dadurch wird eine Seite des Partikels mit einer höheren Geschwindigkeit angeströmt als die gegenüberliegende Seite. Die Folge ist eine ungleichförmige Verteilung des Drucks an der Partikeloberfläche, woraus eine Querkraft resultiert. Diese Querkraft ist nach Saffman benannt, der eine analytische Beziehung für diese Kraft hergeleitet hat [38]:

$$F_S = 1.615 \rho_F \sqrt{\nu_F} d_P^2 v_{rel} \sqrt{\left| \frac{\partial u_F}{\partial y} \right|}, \quad (3.22)$$

wobei $|\partial u_F / \partial y|$ der Betrag des Gradienten der Fluidströmung senkrecht zur mittleren Anströmgeschwindigkeit ist. Die Gültigkeit dieser Beziehung ist auf den Spezialfall einer unbegrenzten linearen Scherströmung beschränkt. Außerdem ist sie nur für einen kleinen Wertebereich der Partikel-Reynoldszahlen gültig, nämlich für:

$$Re_\omega \ll 1 \quad \text{und} \quad Re_P \ll \sqrt{Re_S}, \quad (3.23)$$

wobei Re_ω die Reynoldszahl der Rotation:

$$Re_\omega = \frac{1}{4} \frac{d_P^2 \omega_{rel}}{\nu_F} \quad (3.24)$$

und Re_S die Reynoldszahl der Scherströmung ist:

$$Re_S = \frac{d_P^2 \left| \frac{\partial u_F}{\partial y} \right|}{\nu_F}. \quad (3.25)$$

Für größere Partikel–Reynoldszahlen wurden Korrekturen der Saffman–Beziehung vorgeschlagen [30, 11]. Zur Bestimmung der Saffman–Kraft in komplexen dreidimensionalen Strömungen wird in dieser Arbeit die folgende allgemeinere Bestimmungsgleichung verwendet:

$$\vec{F}_S = \frac{1}{4} \rho_F d_P^2 \sqrt{\nu_F} \frac{1}{\sqrt{\omega_F}} c_S (\vec{v}_{rel} \times \vec{\omega}_F) , \quad c_S = 6.46 . \quad (3.26)$$

Die auf ein Partikel der Masse m_P wirkende Gravitationskraft \vec{F}_G wird berechnet aus:

$$\vec{F}_G = m_P \vec{g} = \frac{\pi}{6} \rho_P d_P^3 \vec{g} , \quad (3.27)$$

wobei \vec{g} die Gravitationsbeschleunigung ist. Die Gravitationskraft hat insbesondere auf die Bewegung relativ großer Partikel einen signifikanten Einfluß, während bei relativ kleinen Partikeln der Einfluß der aerodynamischen Kräfte überwiegt.

Andere Kräfte, wie z.B. die aus einer instationären relativen Beschleunigung des Partikels zum Fluid herrührende sogenannte Basset–Kraft, oder auch die Added–Mass–Kraft, die daraus resultiert, daß das Partikel bei seiner Bewegung Fluidmasse verdrängen und beschleunigen muß, werden hier nicht berücksichtigt. Sie können, wie auch der hydrostatische Auftrieb, vernachlässigt werden. Die Größe dieser Kräfte ist für die hier betrachteten Materialdichten von Partikeln und Gas mit $\rho_P/\rho_F \gg 1$ vernachlässigbar klein gegenüber den oben beschriebenen Kräften.

Damit sind alle relevanten Kräfte beschrieben, die die translatorische Bewegung eines Partikels bestimmen. Setzt man die oben angegebenen Ausdrücke für die verschiedenen Kräfte in die Gleichung (3.10) ein und dividiert die Gleichung durch die Partikelmasse m_P , so erhält man die folgende Differentialgleichung für die Partikelgeschwindigkeit:

$$\begin{aligned} \frac{d}{dt} \vec{v}_P &= \frac{3}{4} \frac{\rho_F}{\rho_P} \frac{1}{d_P} \cdot \\ &\left[c_W v_{rel} \vec{v}_{rel} + c_M \frac{v_{rel}}{\omega_{rel}} (\vec{\omega}_{rel} \times \vec{v}_{rel}) + c_S \frac{2\sqrt{\nu_F}}{\pi\sqrt{\omega_F}} (\vec{v}_{rel} \times \vec{\omega}_F) \right] \\ &+ \vec{g} . \end{aligned} \quad (3.28)$$

Sind die Koordinaten \vec{r}_{P_0} und die Geschwindigkeit \vec{v}_{P_0} des Partikels zu einem Zeitpunkt t_0 bekannt, so ergibt sich die Partikelgeschwindigkeit \vec{v}_{P_1} zum Zeitpunkt $t_1 = t_0 + \Delta t$ durch die numerische Lösung der Gleichung (3.28). Dazu wird hier ein Runge–Kutta–Verfahren verwendet.

Nach Berechnung der neuen Partikelgeschwindigkeit \vec{v}_{P1} kann die Strecke bestimmt werden, die das Partikel während des Zeitschritts Δt zurückgelegt hat. Dazu muß die folgende Differentialgleichung gelöst werden:

$$\frac{d\vec{r}_P}{dt} = \vec{v}_P . \quad (3.29)$$

Hierzu wird angenommen, daß das Partikel in dem Zeitintervall Δt gleichmäßig beschleunigt wird. In diesem Fall ergibt sich der neue Partikelort auf einfache Weise zu:

$$\vec{r}_{P1} = \vec{r}_{P0} + \frac{1}{2} (\vec{v}_{P0} + \vec{v}_{P1}) \cdot \Delta t . \quad (3.30)$$

Damit ist die Berechnung der translatorischen Partikelbewegung vollständig beschrieben, soweit sie durch aerodynamische Kräfte bzw. Körperkräfte hervorgerufen wird. Der Einfluß von Partikel-Wand- und Partikel-Partikel-Kollisionen auf die Partikelbewegung wird weiter unten in separaten Abschnitten behandelt.

Gleichungen für die Partikelrotation

Wie bereits oben dargestellt wurde, kann die Rotation eines Partikels einen großen Einfluß auf die Bewegung des Partikels in einem Fluid haben. So ist z.B. die Größe der Magnus-Kraft und der Saffman-Kraft u.a. abhängig von der Rotationsgeschwindigkeit des Partikels. Der Drehimpuls eines rotierenden Partikels ändert sich, wenn an dem Partikel Momente angreifen. Dieser Zusammenhang wird durch die Drehimpulsgleichung ausgedrückt:

$$I_P \frac{d\vec{\omega}_P}{dt} = \sum_i \vec{T}_i . \quad (3.31)$$

Hierbei ist I_P das Trägheitsmoment des Partikels:

$$I_P = \frac{1}{10} m_P d_P^2 = \frac{\pi}{60} \rho_P d_P^5 , \quad (3.32)$$

und $\sum_i \vec{T}_i$ ist die Summe der an dem Partikel angreifenden Momente. Diese Momente können z.B. aerodynamischer Natur sein, d.h. durch die Reibung zwischen der Partikeloberfläche und dem umgebenden Fluid entstehen. Nach Untersuchungen von Dennis [12] kann das vom Fluid auf das Partikel ausgeübte Moment in folgender Weise bestimmt werden:

$$\vec{T} = \frac{\rho_F}{2} \left(\frac{d_P}{2} \right)^5 c_\omega \omega_{rel} \vec{\omega}_{rel} . \quad (3.33)$$

Die Größe des in dieser Gleichung verwendeten Rotationsbeiwertes c_ω ist abhängig von der Reynoldszahl der Rotation Re_ω . Diese Abhängigkeit ist in der Tabelle 3.2

angegeben sowie in der Abbildung 3.4 graphisch dargestellt. Neben den aerodynamischen Kräften treten bei Kollisionen mit Wänden oder anderen Partikeln tangentiale Kräfte auf, die ein Moment bezüglich des Partikelmittelpunktes hervorrufen. Die Wirkung dieser Kräfte wird am Ende dieses Abschnitts bzw. im Kapitel 4 beschrieben.

Nach Einsetzen der Gleichungen (3.32) und (3.33) in die Gleichung (3.31) und Division durch I_P ergibt sich die Differentialgleichung für die Rotationsgeschwindigkeit des Partikels zu:

$$\frac{d}{dt}\vec{\omega}_P = \frac{15}{16\pi} \frac{\rho_F}{\rho_P} c_\omega \omega_{rel} \vec{\omega}_{rel} . \quad (3.34)$$

Diese Gleichung wird mit Hilfe eines Runge–Kutta–Algorithmus numerisch gelöst. Im Unterschied zur Berechnung der translatorischen Bewegung erfolgt hier keine weitere Integration zur Bestimmung des Winkels, um den sich das Partikel während des Zeitschritts Δt gedreht hat. Dieser ist hier nicht von Interesse, da die Partikel als kugelförmig angenommen werden.

Einfluß der Fluidturbulenz auf die Partikelbewegung

In den Gleichungen, die in den vorhergehenden Unterabschnitten für die Berechnung der Partikelbewegung angegeben wurden, wurde für die Fluidgeschwindigkeit jeweils die zeitlich gemittelte Geschwindigkeit verwendet. Diese ergibt sich aus der Berechnung der turbulenten Fluidströmung unter Verwendung des k – ε –Modells zur Modellierung der Turbulenz. Die turbulenten Schwankungsbewegungen in der Fluidströmung können jedoch die Partikelbewegung signifikant beeinflussen. Insbesondere verstärken sie die Dispersion relativ kleiner Partikel.

Um diesen Effekt zu berücksichtigen, wird hier das Lagrangesche Stochastisch–Deterministische (LSD) Modell nach Milojević [31] verwendet. Hierbei wird in den Bewegungsgleichungen für das Partikel die mittlere Fluidgeschwindigkeit u_F durch die momentane Geschwindigkeit $u_{F,akt}$ ersetzt:

$$u_{F,akt} = u_F + u'_F . \quad (3.35)$$

Die Schwankungsgeschwindigkeit u'_F wird dabei mit Hilfe eines stochastischen Verfahrens bestimmt. Dazu wird angenommen, daß u'_F eine Zufallsgröße ist, die einer Gaußverteilung mit dem Mittelwert 0 m/s und der Standardabweichung $\sqrt{\frac{2}{3}k}$ unterliegt, wobei k die Turbulenzenergie des Fluides ist. Nach Bestimmung der Schwankungsgeschwindigkeit wird in den Bewegungsgleichungen für das Partikel die mittlere Fluidgeschwindigkeit durch die momentane Geschwindigkeit ersetzt und so die Partikelbewegung unter dem Einfluß der Turbulenz berechnet.

Re_ω	c_ω
$0 < Re_\omega \leq 10$	$\frac{16\pi}{Re_\omega}$
$10 < Re_\omega \leq 1000$	$\frac{6.45}{\sqrt{Re_\omega}} + \frac{32.1}{Re_\omega}$
$1000 < Re_\omega \leq 40000$	$\frac{6.8}{\sqrt{Re_\omega}}$
$40000 < Re_\omega \leq 400000$	$\frac{0.058}{\sqrt[20]{Re_\omega}}$
$400000 < Re_\omega \leq 10^7$	$\frac{0.397}{\sqrt[5]{Re_\omega}}$

Tabelle 3.2: Abhängigkeit des Rotationsbeiwertes von der Reynoldszahl der Rotation

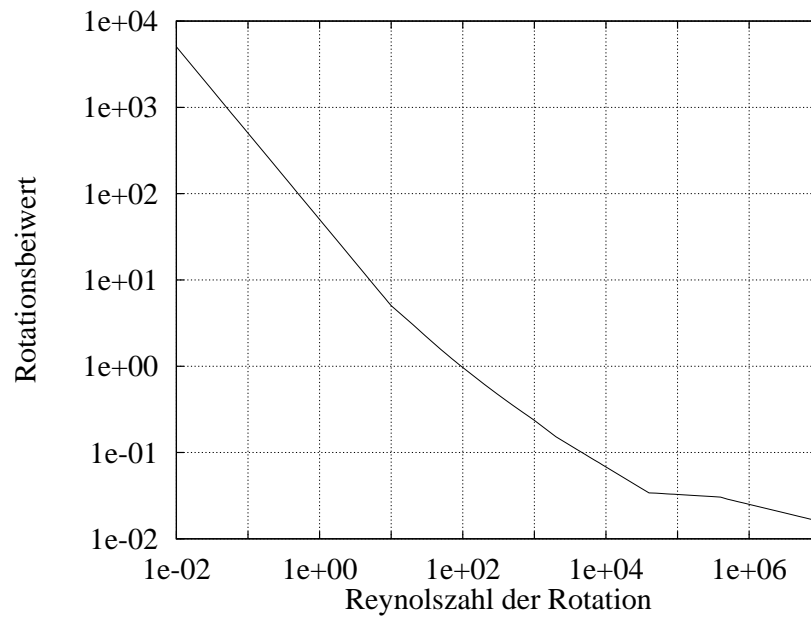


Abbildung 3.4: Verlauf des Rotationsbeiwertes c_ω in Abhängigkeit von der Reynoldszahl der Rotation Re_ω

Das Erzeugen einer zufälligen Schwankungsgeschwindigkeit ist gleichbedeutend mit dem Generieren eines Turbulenzwirbels, den das Partikel auf seiner Bahn durchquert. Die Länge des Zeitintervalls Δt_w , in dem das Partikel mit dem Wirbel in Wechselwirkung steht, ist dabei durch zwei Faktoren begrenzt. Diese sind zum einen die Wirbellebensdauer T_E und zum anderen die Durchgangszeit T_D , d.h. die Zeit, die das Partikel benötigt, um den Wirbel zu durchqueren. Zur Bestimmung der Wirbellebensdauer wird folgende Beziehung verwendet:

$$T_E = c_T \frac{k}{\varepsilon}, \quad \text{mit } c_T = 0.3. \quad (3.36)$$

Die Durchgangszeit T_D hängt ab von der Größe des Wirbels L_E und der Relativgeschwindigkeit zwischen Wirbel und Partikel. Die Größe des Wirbels wird bestimmt aus:

$$L_E = T_E \sqrt{\frac{2}{3}k}, \quad (3.37)$$

und somit ergibt sich die Durchgangszeit zu:

$$T_D = \frac{L_E}{v_{rel}}. \quad (3.38)$$

Ist ein turbulenter Wirbel generiert worden, so beträgt die Wechselwirkungszeit des Partikels mit diesem Wirbel:

$$\Delta t_w = \min(T_E, T_D). \quad (3.39)$$

Nach Ablauf dieser Zeit wird ein neuer Wirbel generiert, mit dem das Partikel in Wechselwirkung tritt.

Einfluß der Partikel auf die Fluidbewegung

In den vorangegangenen Unterabschnitten wurde beschrieben, wie die Partikelbewegung durch die aerodynamischen Kräfte des Fluides bestimmt wird. Wenn sich durch die Krafteinwirkung der Fluidströmung der Impuls eines Partikels um $\Delta \vec{p}_P$ ändert, so muß sich aus Gründen der Impulserhaltung der Impuls des Fluides in der Umgebung des Partikels ebenfalls ändern, und zwar in folgender Weise:

$$\Delta \vec{p}_F = -\Delta \vec{p}_P. \quad (3.40)$$

Dies hat zur Folge, daß die Fluidströmung durch die Bewegung der Partikel beeinflusst wird. Es findet eine Wechselwirkung zwischen der dispersen und der fluiden Phase statt, die auf einer gegenseitigen Impulsübertragung zwischen den Phasen beruht.

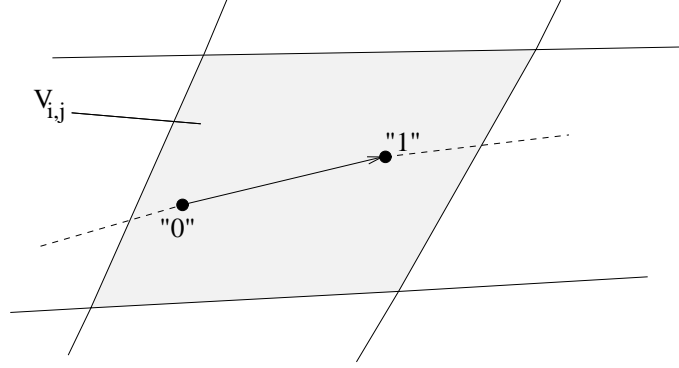


Abbildung 3.5: Schema eines Partikelschrittes in einem Kontrollvolumen

Zur Berücksichtigung dieser Phasenkopplung in der numerischen Berechnung der Gas-Partikel-Strömung wurde von Crowe [7] die PSI-Cell (Particle-Source-In-Cell) Methode entwickelt. Hierbei wird der Einfluß der Partikel auf die Fluidströmung durch Quellterme modelliert, die in die Impuls- sowie die k - und ε -Gleichungen des Fluides eingeführt werden (Gleichungen (2.3), (2.6) und (2.7)).

In der Abbildung 3.5 ist schematisch die Bewegung eines Partikels von einem Punkt "0" zu einem Punkt "1" innerhalb eines Kontrollvolumens dargestellt. Unter der Annahme, daß die Änderung der Partikelgeschwindigkeit zwischen diesen Punkten linear verläuft, ergeben sich die Impulsquellterme, die aus diesem Partikelschritt resultieren, für das betrachtete Kontrollvolumen zu:

$$Q_{Pu\,i,j} = -\frac{1}{V_{i,j}} m_P \left(\frac{u_P^1 - u_P^0}{\Delta t^1} - g_x \right), \quad (3.41)$$

$$Q_{Pv\,i,j} = -\frac{1}{V_{i,j}} m_P \left(\frac{v_P^1 - v_P^0}{\Delta t^1} - g_y \right). \quad (3.42)$$

Hierbei ist $V_{i,j}$ die Größe des Kontrollvolumens und Δt^1 die für die Bewegung von "0" nach "1" benötigte Zeit. Zur Bestimmung von Quelltermen für die k - ε -Gleichungen, d.h. zur Berücksichtigung des Einflusses der dispersen Phase auf die Fluidturbulenz, sind bisher keine allgemeingültigen Berechnungsvorschriften bekannt. Das Vorhandensein von Partikeln kann in Abhängigkeit von der Massenbeladung und dem Partikeldurchmesser die Turbulenz sowohl anregen als auch dämpfen. Untersuchungen hierzu wurden z.B. von Shuen et al. [40] und Squires und Eaton [43] durchgeführt.

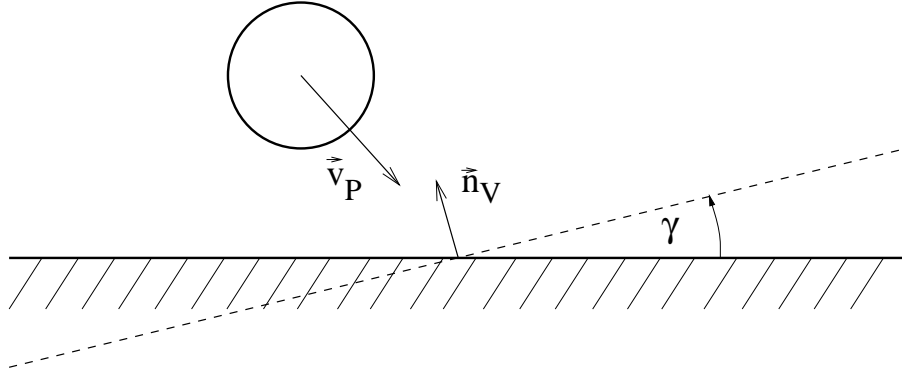


Abbildung 3.6: Stoß des Partikels mit einer virtuellen Wand

Partikel–Wand–Kollisionen

Die meisten Strömungen in der Energie- und Verfahrenstechnik sind Strömungen innerhalb von Anlagen oder Apparaten. Diese Strömungen werden in der Regel von festen Wänden begrenzt. Betrachtet man Gas-Partikel-Strömungen, so hat die Wechselwirkung der Partikel mit den Wänden einen großen Einfluß auf das Strömungsverhalten. Dies gilt insbesondere für relativ große Partikel, die aufgrund ihrer Massenträgheit verhältnismäßig langsam auf Änderungen im Geschwindigkeitsfeld des Gases reagieren und daher oft mit den Wänden kollidieren. Ein besonders wichtiger Parameter bei den Partikel-Wand-Kollisionen ist die Wandrauhigkeit, die aus dem Herstellungsprozeß des Wandbauteils herrührt. Die Abmessungen der Unebenheiten der Wand liegen oft in derselben Größenordnung wie der Partikeldurchmesser. Dies hat zur Folge, daß ein Partikel, das mit der Wand kollidiert, lokal eine andere Wandneigung antrifft, als es bei einer ideal glatten Wand der Fall wäre. Dadurch wird eine stärkere Dispersion der Partikelphase verursacht.

Zur Modellierung der Wandrauhigkeit wurden eine Reihe verschiedener Ansätze entwickelt, so z.B. von Matsumoto und Saito [27, 28, 29], Tsuji et al. [50] und Frank [15]. Das in dieser Arbeit verwendete Wandrauhigkeitsmodell basiert auf einem Modell von Sommerfeld [41]. Hierbei stößt das Partikel mit einer virtuellen Wand, die gegenüber der mittleren Wandebene um einen Winkel γ angestellt ist (siehe Abbildung 3.6). Der Inklinationswinkel γ ist eine Zufallsgröße, die aus einer Gaußverteilung mit dem Mittelwert $\gamma = 0$ und der Standardabweichung $\Delta\gamma$ bestimmt wird. Die Größe der Standardabweichung ist von den Rauigkeitsparametern der Wand und dem Durchmesser des auftreffenden Partikels abhängig. Wie man anhand der Abbildungen 3.7 und 3.8 sieht, ist der Stoßwinkel für kleine Partikel in der Regel größer als für große Partikel. Partikel, deren Durchmesser deutlich kleiner als der Abstand

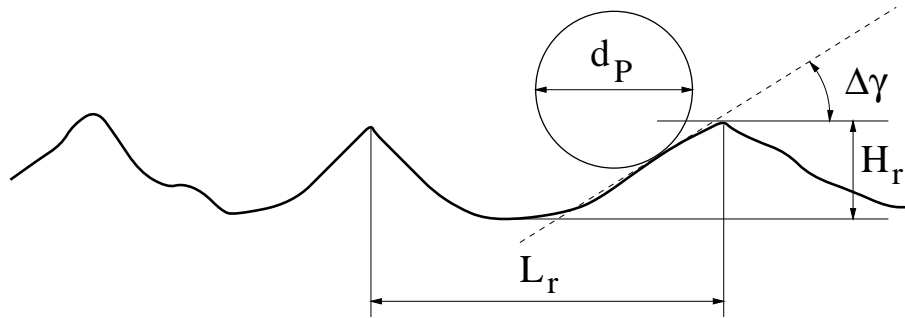


Abbildung 3.7: Rauhwinkel $\Delta\gamma$ für kleine Partikel

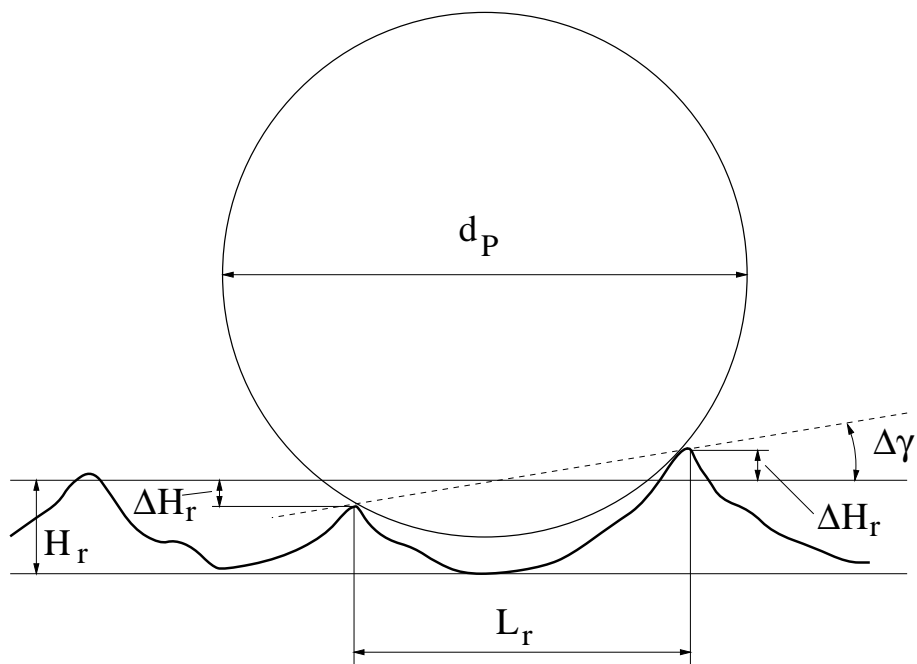


Abbildung 3.8: Rauhwinkel $\Delta\gamma$ für große Partikel

L_r zwischen zwei Rauigkeitsspitzen ist, stoßen überwiegend mit den Flanken der Rauigkeitserhebungen. Dagegen stoßen wesentlich größere Partikel hauptsächlich mit den Spitzen der Rauigkeitserhebungen. Aus geometrischen Betrachtungen läßt sich die Standardabweichung des Inklinationswinkels in folgender Weise abschätzen:

$$\Delta\gamma = \begin{cases} \arctan \frac{2H_r}{L_r} & \text{für } d_P < \frac{L_r}{\sin\left(\arctan \frac{2H_r}{L_r}\right)} \\ \arctan \frac{2\Delta H_r}{L_r} & \text{für } d_P \geq \frac{L_r}{\sin\left(\arctan \frac{2H_r}{L_r}\right)} \end{cases} \quad (3.43)$$

Hierin ist L_r die mittlere Rauigkeitslänge, d.h. der mittlere Abstand zwischen zwei Rauigkeitsspitzen, H_r die mittlere Rauigkeitstiefe und ΔH_r die Standardabweichung von H_r . Diese Parameter müssen experimentell ermittelt werden.

Nachdem der Inklinationswinkel und damit die Lage der virtuellen Wand bestimmt worden ist, wird überprüft, ob sich das Partikel tatsächlich auf physikalisch sinnvolle Weise an die zufällig erzeugte Wand annähert. Dazu betrachtet man den Geschwindigkeitsvektor des Partikels \vec{v}_P und den ins Innere des Strömungsgebietes gerichteten Normalenvektor der virtuellen Wand \vec{n}_v . Ein physikalisch sinnvoller Stoß findet nur dann statt, wenn:

$$\vec{v}_P \cdot \vec{n}_v \leq 0 . \quad (3.44)$$

Im anderen Fall wird die aktuelle Wand verworfen und eine neue virtuelle Wand generiert.

Die Translations- und Rotationsgeschwindigkeit des Partikels nach dem Wandstoß ergeben sich aus den Impuls- bzw. Drehimpulsgleichungen. Die Berechnung des Stoßvorgangs erfolgt in einem wandorientierten Koordinatensystem. Die y -Achse des Koordinatensystems ist hierbei identisch mit dem nach innen gerichteten Normalenvektor der virtuellen Wand. In der Abbildung 3.9 ist eine solche Stoßkonfiguration mit den im folgenden verwendeten Bezeichnungen dargestellt. Nach Tsuji [50] kann man zwei verschiedene Arten von Wandstößen unterscheiden, den Gleitstoß und den Haftstoß. Im ersten Fall gleitet das Partikel während des gesamten Stoßvorgangs an der Wand entlang. Im zweiten Fall kommt das Gleiten vor dem Ende des Stoßvorgangs zum Erliegen. Die Beziehungen zwischen den Größen vor und nach dem Stoß lauten für diese beiden Fälle:

$$1. \text{ Gleitstoß für } -\frac{2}{7 f_w (e_w + 1)} \leq \frac{v_p^{(1)}}{|v_r|} \leq 0 :$$

$$u_p^{(2)} = u_p^{(1)} + f_w (e_w + 1) v_p^{(1)} ,$$

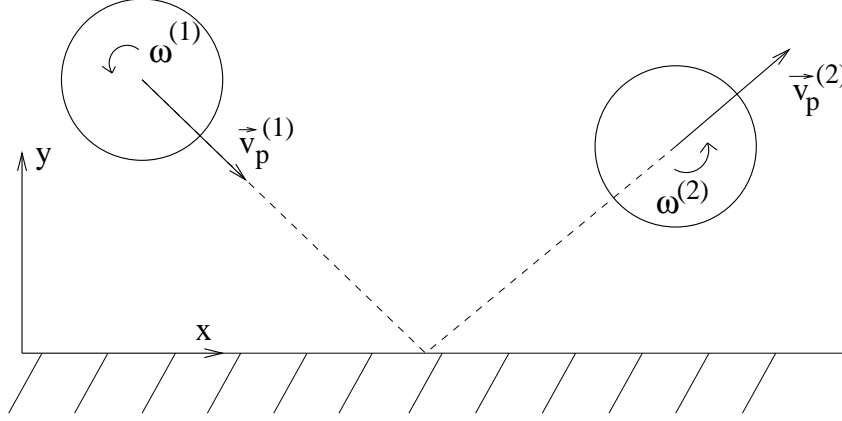


Abbildung 3.9: Bezeichnungen zur Berechnung des Partikel–Wand–Stoßes

$$\begin{aligned} v_p^{(2)} &= -e_w v_p^{(1)} \quad , \\ \omega^{(2)} &= \omega^{(1)} + \frac{5}{d_p} f_w (e_w + 1) v_p^{(1)} \quad , \end{aligned} \quad (3.45)$$

2. Haftstoß für $\frac{v_p^{(1)}}{|v_r|} < -\frac{2}{7 f_w (e_w + 1)} \quad :$

$$\begin{aligned} u_p^{(2)} &= \frac{5}{7} \left(u_p^{(1)} - \frac{d_p}{5} \omega^{(1)} \right) \quad , \\ v_p^{(2)} &= -e_w v_p^{(1)} \quad , \\ \omega^{(2)} &= -\frac{2}{d_p} u_p^{(1)} \quad , \end{aligned} \quad (3.46)$$

mit

$$v_r = u_p^{(1)} + \frac{d_p}{2} \omega^{(1)} \quad . \quad (3.47)$$

Das Superskript ⁽¹⁾ bezeichnet hierbei jeweils Größen vor dem Stoß und das Superskript ⁽²⁾ bezeichnet Größen nach dem Stoß. Der Parameter f_w ist der Gleitreibungsbeiwert und e_w die Stoßzahl für die aktuelle Materialpaarung von Partikel und Wand, und die Größe v_r ist die Relativgeschwindigkeit zwischen der Partikeloberfläche und der Wand am Punkt des Wandkontaktes.

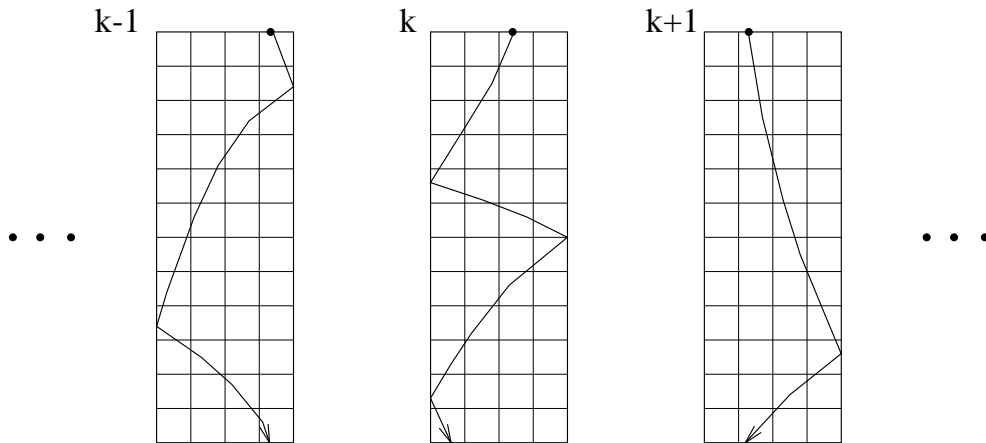


Abbildung 3.10: Sukzessive Trajektorienberechnung am Beispiel eines vertikalen Kanals

3.3.2 Berechnung der Strömung der dispersen Phase

Im vorangegangenen Abschnitt 3.3.1 wurde beschrieben, wie die Bewegung eines einzelnen Partikels in einer Gas-Partikel-Strömung berechnet wird. Aus der Summe der einzelnen Partikelbewegungen in einer solchen Strömung resultieren die makroskopischen Strömungsgrößen der Partikelphase, wie z.B. die lokale Partikelkonzentration oder die mittlere lokale Partikelgeschwindigkeit. Zur Bestimmung dieser Größen existieren im wesentlichen zwei verschiedene Verfahren, die im folgenden beschrieben werden.

Trajektorienberechnung

Dieses Verfahren beruht auf der aufeinanderfolgenden Berechnung einer großen Anzahl von Partikeltrajektorien (siehe Abbildung 3.10). Eine Trajektorie ist hierbei die Bahn eines Partikels von einem Startpunkt, der in der Regel an einer Eintrittsöffnung liegt, bis zum Erreichen eines Abbruchkriteriums, das z.B. das Erreichen einer Austrittsöffnung sein kann. Es wird angenommen, daß jedes Partikel, das am selben Startpunkt und mit denselben Anfangsbedingungen (Geschwindigkeit, Rotationsgeschwindigkeit, Durchmesser, usw.) in das Strömungsgebiet eintritt, dieselbe Bahn durchläuft. Die Voraussetzung für die Anwendung dieses Verfahrens ist, daß das komplette Gasgeschwindigkeitsfeld vor Beginn der Partikelberechnung bekannt ist und daß die gesamte Strömung stationär ist. Eine Trajektorie repräsentiert somit einen Strom von Partikeln, die alle dieselben physikalischen Eigenschaften besit-

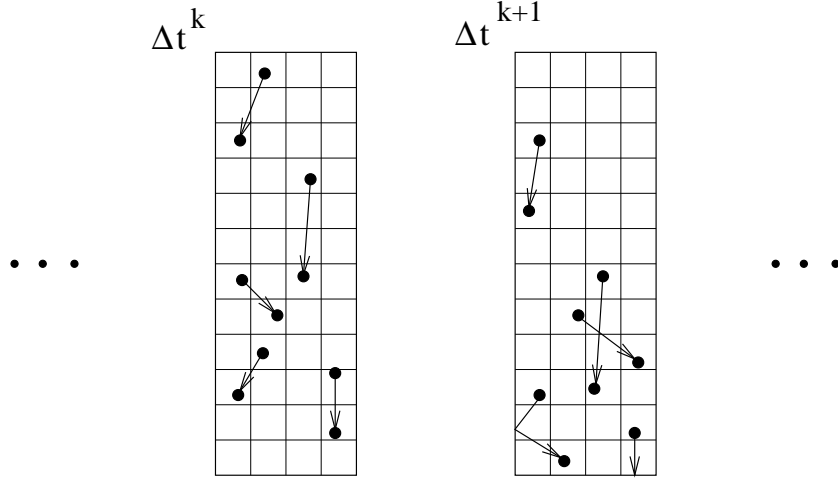


Abbildung 3.11: Simultane Partikelverfolgung am Beispiel eines vertikalen Kanals

zen. Dieser Strom ist ein Teil des Partikelmassenstroms in einer stationären Gas-Partikel-Strömung. Die Anzahl von Partikeln pro Zeiteinheit \dot{N}_P , die sich entlang einer Trajektorie bewegen, ergibt sich aus dem Gesamtpartikelmassenstrom \dot{m}_P und der Anzahl der Trajektorien N_T zu:

$$\dot{N}_P = \frac{\dot{m}_P}{\rho_P \sum_{k=1}^{N_T} \frac{\pi}{6} d_{P_k}^3} . \quad (3.48)$$

Die makroskopischen Strömungsgrößen der Partikelphase ergeben sich für jedes Kontrollvolumen aus der Summe der dieses Kontrollvolumen durchquerenden Trajektorien. Um für diese Größen statistisch gesicherte Werte zu erhalten, muß die Anzahl der berechneten Trajektorien ausreichend groß sein.

Simultane Partikelverfolgung

In diesem Verfahren wird die Bewegung aller simulierten Partikel, die sich zu einem bestimmten Zeitpunkt innerhalb eines Strömungsgebietes aufhalten, simultan berechnet (siehe Abbildung 3.11). Um den Rechen- und Speicheraufwand zu begrenzen, ist dabei die Anzahl der simulierten Partikel im allgemeinen kleiner als die Anzahl der Partikel in der realen Strömung. In diesem Fall repräsentiert ein simuliertes Partikel eine bestimmte Anzahl realer Partikel mit denselben physikalischen Eigenschaften.

Wie in der Abbildung 3.11 schematisch dargestellt ist, wird innerhalb eines Zeitschrittes Δt jeweils die Bewegung aller simulierten Partikel berechnet. Kollidiert ein Partikel während dieses Zeitschritts mit der Strömungsberandung, so erfolgt die Berechnung des Schrittes in zwei Teilschritten. Zunächst wird die Partikelbewegung bis zum Erreichen der Wand berechnet. Mit Hilfe des im Abschnitt 3.3.1 beschriebenen Kollisionsmodells werden die Partikelgeschwindigkeit und –rotationsgeschwindigkeit nach dem Stoß bestimmt, und unter Verwendung dieser Größen wird der Zeitschritt zu Ende geführt. Nach dem Abschluß der Berechnung aller einzelnen Partikelschritte des aktuellen Zeitschrittes wird der nächste Zeitschritt in gleicher Weise ausgeführt.

Das Verfahren der simultanen Partikelverfolgung ist ein inhärent instationäres Berechnungsverfahren. Die makroskopischen Strömungsgrößen der Partikelphase ergeben sich für ein Kontrollvolumen zu jedem Zeitpunkt aus der Mittelwertbildung über alle Partikel, die sich zu diesem Zeitpunkt in dem betreffenden Kontrollvolumen befinden. Das Verfahren eignet sich sowohl für die Simulation instationärer als auch makroskopisch stationärer Gas–Partikel–Strömungen. Insbesondere ist es auch für die Simulation kollisionsbehafteter Strömungen geeignet, was im Kapitel 4 ausführlich erläutert wird.

Kapitel 4

Simulation von Partikel–Partikel–Kollisionen

In diesem Kapitel soll die Behandlung der Partikel–Partikel–Kollisionen im Rahmen der Lagrangeschen Berechnung der Partikelbewegung vorgestellt werden. Im Abschnitt 4.1 wird zunächst anhand einiger Beispiele erläutert, welchen Einfluß die Kollisionen auf das Verhalten von Gas–Feststoff–Strömungen haben. Anschließend sollen im Abschnitt 4.2 einige Begriffe erklärt und Definitionen angegeben werden, die in diesem Zusammenhang wichtig sind und im weiteren Verlauf des Kapitels häufig verwendet werden. Die Berechnung der translatorischen und rotatorischen Geschwindigkeit zweier Partikel nach einer Kollision wird im Abschnitt 4.3 beschrieben. Diese Berechnungsvorschriften werden in allen Kollisionsmodellen gleichermaßen verwendet. Im Abschnitt 4.4 erfolgt schließlich die Beschreibung der verschiedenen Modelle, die in den vergangenen Jahren zur Simulation der Inter–Partikel–Kollisionen entwickelt wurden.

4.1 Bedeutung der Partikel–Partikel–Kollisionen

Der Einfluß, den die Partikel–Partikel–Kollisionen auf das Verhalten einer mehrphasigen Strömung haben, wurde in der Vergangenheit von verschiedenen Autoren untersucht. So führte z.B. Tanaka [45] Messungen an einer Strömung von Polystyrenpartikeln und Luft in einem vertikalen Rohr durch. Hierbei zeigte sich, daß bei sehr geringen Beladungen die Partikelkonzentration nahe der Rohrachse am größten ist und nach außen hin stark abnimmt. Bereits bei einer moderaten volumetrischen Beladung von $\alpha_P = 4 \cdot 10^{-4}$, die einer Massenbeladung von $\eta \approx 0.35$ entspricht, verursachen die Inter–Partikel–Kollisionen eine verstärkte Dispersion der Partikel, was

zu einer gleichmäßigeren Konzentrationsverteilung im Rohrquerschnitt führt. Diese Messungen wurden durch entsprechende Simulationen von Tanaka und Tsuji [46] bestätigt.

Tashiro et al. [48] untersuchten ebenfalls numerisch eine vertikale Rohrströmung. Hierbei wurden Partikel mit einem Durchmesser von $d_p = 2 - 4$ mm und einer Dichte von $\rho_p = 1000 \text{ kg/m}^3$ betrachtet, die in einem Luftstrom transportiert wurden. Dabei zeigte sich, daß sich bei einer Massenbeladung von $\eta = 0.642$ durch den Einfluß der Kollisionen eine nahezu konstante Konzentrationsverteilung über den Rohrquerschnitt einstellt. Im Gegensatz dazu ergab sich – analog zu den Ergebnissen Tanakas [45, 46] – ohne Berücksichtigung der Kollisionen eine sehr hohe Partikelkonzentration in der Nähe der Rohrachse, die nach außen hin stark abnimmt. Des weiteren zeigte sich, daß die axiale Partikelgeschwindigkeit im kollisionsbehafteten Fall ebenfalls sehr gleichmäßig über den Rohrquerschnitt verteilt ist. Dies ist eine Folge des durch die Kollisionen induzierten Impulsaustausches zwischen den Partikeln. Im kollisionsfreien Fall bewegen sich Partikel, die sich in der Mitte des Rohres aufhalten, aufgrund der höheren lokalen Gasgeschwindigkeit wesentlich schneller in axialer Richtung als Partikel, die einen größeren Abstand zur Rohrachse haben.

Oesterlé und Petitjean [34] führten numerische Simulationen einer horizontalen Rohrströmung durch. Hierbei wurde der pneumatische Transport von Glaspartikeln mit einem mittleren Durchmesser von $d_p = 100 \text{ } \mu\text{m}$ in einem Rohr mit einem Durchmesser von 30 mm untersucht. Bei einer geringen Massenbeladung ist dabei die vertikale Konzentrationsverteilung der Partikel relativ gleichmäßig. Dies ist auf den im Abschnitt 3.3.1 beschriebenen Einfluß der Wandrauhigkeit zurückzuführen. Mit zunehmender Massenbeladung nimmt jedoch die Partikelkonzentration im unteren Teil des Rohrquerschnittes zu. Dies ist damit zu erklären, daß die von der rauhen Wand zurückprallenden Partikel nicht mehr so weit in das Strömungsgebiet eindringen können, da sie auf ihrem Weg mit anderen Partikeln kollidieren. Bis zu einer Massenbeladung von $\eta = 10$ ist zu beobachten, daß die maximale Konzentration nicht unmittelbar an der Wand vorliegt, sondern ca. 4.5 mm von der Wand entfernt. Dies wird von Oesterlé und Petitjean damit erklärt, daß in unmittelbarer Wandnähe sehr große Unterschiede in den Geschwindigkeiten der auftreffenden und rückprallenden Partikeln existieren, was eine große Anzahl von Inter-Partikel-Kollisionen zur Folge hat. Dadurch entsteht ein abschirmender Effekt, durch den die Partikelkonzentration in unmittelbarer Wandnähe sinkt. Bei einer Massenbeladung von $\eta = 20$ ist dieser Effekt allerdings nicht mehr zu beobachten. In diesem Fall nimmt die Konzentration nach unten hin stark zu und hat ihr Maximum direkt an der Rohrwand. Eine Erklärung für das Entfallen dieses Effekts bei sehr hohen Beladungen wird in [34] nicht angegeben.

Der Einfluß der Partikel-Partikel-Kollisionen auf die Erosion der Strömungsberandung wurde z.B. in Experimenten von Umois und Kleis [53] sowie in entspre-

chenden Simulationen von Kitron et al. [24] untersucht. Hierbei wurde gezeigt, daß die Kollisionen die Erosion sowohl verstärken als auch abschwächen können. Der verstärkende Charakter wird dadurch erklärt, daß die Partikel in einer kollisionsbehafteten Strömung unter anderen Winkeln und mit anderen Geschwindigkeiten auf die Wandoberfläche auftreffen als in einer kollisionsfreien Strömung. Dagegen können bei hohen Beladungen die zurückprallenden Partikel die Wand vor den auftreffenden Partikeln abschirmen. Dadurch verringert sich die Anzahl und die Geschwindigkeit der eintreffenden Partikel, was wiederum den Materialabtrag von der Wand verringert.

Anhand dieser Beispiele sieht man, daß die Inter-Partikel-Kollisionen schon bei moderaten Beladungen den Charakter einer Gas-Feststoff-Strömung entscheidend prägen können. Es ist daher wichtig, diese Kollisionen bei der numerischen Simulation solcher Strömungen zu berücksichtigen.

4.2 Allgemeine Definitionen und Begriffe

Eine wichtige Größe im Zusammenhang mit Partikel-Partikel-Kollisionen und deren Modellierung ist die Anzahldichte der Partikel n_P , d.h. die Anzahl der Partikel, die sich in einem Einheitsvolumen des Gas-Feststoff-Gemisches aufhalten. Wie bereits im Abschnitt 3.1 erläutert wurde, ist die Angabe einer solchen Größe nur dann sinnvoll, wenn sich in einem Kontrollvolumen, das wesentlich kleiner als die charakteristische Abmessung des Strömungsgebietes ist, hinreichend viele Partikel aufhalten. Ist diese Voraussetzung erfüllt, so beträgt das mittlere Volumen Ω_P , das jedem Partikel zur Verfügung steht:

$$\Omega_P = \frac{1}{n_P} , \quad (4.1)$$

und der mittlere Abstand zwischen den Partikeln δ_P läßt sich abschätzen mit:

$$\delta_P = n_P^{-\frac{1}{3}} . \quad (4.2)$$

In dieser Arbeit sollen Strömungen betrachtet werden, die eine mittlere bis hohe Massenbeladung aufweisen, deren volumetrische Beladung aber dennoch klein ist. Daß diese Annahme plausibel ist, sieht man z.B. daran, daß in einem Gemisch von Kohlepartikeln und Luft mit einer relativ hohen Massenbeladung von $\eta = 10$ die Volumenbeladung des Gemisches $\alpha_P \approx 5 \cdot 10^{-3}$ beträgt. In solchen Strömungen ist der mittlere Abstand zwischen den Partikeln sehr viel größer als der Partikeldurchmesser:

$$\delta_P \gg d_P . \quad (4.3)$$

Dies hat zur Folge, daß ein Partikel sich die meiste Zeit außerhalb des Einflusses anderer Partikel bewegt, d.h. daß ein Stoß mit einem anderen Partikel ein relativ seltenes Ereignis ist. Des weiteren kann man davon ausgehen, daß an der überwiegenden Mehrzahl der Stöße nur jeweils zwei Partikel beteiligt sind. Aus diesem Grund sollen hier ausschließlich binäre Stöße betrachtet werden.

Die mittlere Strecke, die ein Partikel zwischen zwei aufeinanderfolgenden Kollisionen zurücklegt, wird als mittlere freie Weglänge λ_P bezeichnet. Die Zeit, die das Partikel dazu benötigt, ist die mittlere Kollisionszeit τ_C . Diese Größe wurde bereits im Abschnitt 3.1 vorgestellt. Durch den Vergleich von τ_C mit der aerodynamischen Relaxationszeit τ_A läßt sich entscheiden, ob die Partikelbewegung überwiegend von den aerodynamischen Kräften des Fluides oder von den Inter-Partikel-Kollisionen dominiert wird. Der Kehrwert der Kollisionszeit ist die mittlere Kollisionsrate oder Kollisionsfrequenz ν_C , d.h. die mittlere Anzahl von Kollisionen, die ein Partikel pro Zeiteinheit ausführt:

$$\nu_C = \frac{1}{\tau_C} . \quad (4.4)$$

Damit sind die wichtigsten Größen definiert, die zur Beschreibung, Modellierung und Berechnung von Partikel-Partikel-Kollisionen benötigt werden.

4.3 Berechnung der Kollision zweier Partikel

Die verschiedenen Modelle zur Simulation von Partikel-Partikel-Kollisionen unterscheiden sich in den Verfahren zur Detektion einer Kollision sowie zur Bestimmung der daran beteiligten Kollisionspartner. Allen Modellen ist jedoch gemeinsam, daß die Geschwindigkeiten und Rotationsgeschwindigkeiten der beteiligten Partikel nach dem Stoß mit Hilfe der Impuls- bzw. Drehimpulsgleichungen aus den entsprechenden Größen vor dem Stoß ermittelt werden. Bevor die Beschreibung der verschiedenen Kollisionsmodelle erfolgt, soll daher im folgenden die Berechnung der translatorischen und rotatorischen Geschwindigkeiten zweier Partikel nach einem Stoß beschrieben werden. Die dabei angegebenen Gleichungen sind aus [46] entnommen, jedoch werden sie in allen Kollisionsmodellen gleichermaßen verwendet.

Die Anwendung der Impuls- und Drehimpulsgleichungen auf den Stoß zwischen den Partikeln i und j ergibt für die Größen nach dem Stoß:

$$\vec{v}_{Pi}^{(2)} = \vec{v}_{Pi}^{(1)} + \frac{\vec{J}}{m_{Pi}} , \quad (4.5)$$

$$\vec{v}_{Pj}^{(2)} = \vec{v}_{Pj}^{(1)} - \frac{\vec{J}}{m_{Pj}} , \quad (4.6)$$

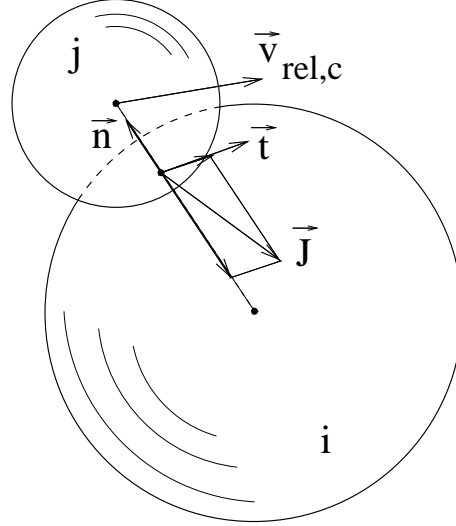


Abbildung 4.1: Räumliche Darstellung der Kollision zweier Partikel

$$\vec{\omega}_{P_i}^{(2)} = \vec{\omega}_{P_i}^{(1)} + \frac{1}{I_{P_i}} \frac{d_{P_i}}{2} (\vec{n} \times \vec{J}) , \quad (4.7)$$

$$\vec{\omega}_{P_j}^{(2)} = \vec{\omega}_{P_j}^{(1)} + \frac{1}{I_{P_j}} \frac{d_{P_j}}{2} (\vec{n} \times \vec{J}) . \quad (4.8)$$

Hierin bezeichnen die Superskripte ⁽¹⁾ und ⁽²⁾ die Größen vor bzw. nach dem Stoß, I_P ist das Trägheitsmoment des Partikels, und \vec{n} ist der vom Partikel i zum Partikel j gerichtete Normalenvektor, wie in der Abbildung 4.1 gezeigt. Der Vektor \vec{n} steht senkrecht auf der Oberfläche beider Partikel in deren Berührungspunkt und läßt sich bestimmen aus:

$$\vec{n} = \frac{\vec{r}_c}{\frac{d_{P_i}}{2} + \frac{d_{P_j}}{2}} , \quad (4.9)$$

wobei \vec{r}_c der Vektor vom Mittelpunkt des Partikels i zum Mittelpunkt des Partikels j ist. Die Größe \vec{J} ist der während der Kollision auf das Partikel i ausgeübte “Kraftimpuls“. Da der Kollisionsvorgang eine endliche Zeitdauer hat, ist der Kraftimpuls identisch mit dem Zeitintegral der Kraft \vec{F}_C , die während der Kollision auf das Partikel i wirkt:

$$\vec{J} \equiv \int_{t^{(1)}}^{t^{(2)}} \vec{F}_C dt . \quad (4.10)$$

Der Kraftimpuls läßt sich in Komponenten normal bzw. tangential zur Partikel-

oberfläche zerlegen:

$$\vec{J} = J_n \vec{n} + J_t \vec{t} . \quad (4.11)$$

Mit Hilfe der Relativgeschwindigkeit zwischen den Partikelmittelpunkten:

$$\vec{v}_{rel,c} = \vec{v}_{P_j}^{(1)} - \vec{v}_{P_i}^{(1)} , \quad (4.12)$$

und der relativen Gleitgeschwindigkeit im Berührungspunkt zwischen den Partikeloberflächen :

$$\vec{v}_{fc} = \vec{v}_{rel,c} - (\vec{v}_{rel,c} \cdot \vec{n}) \vec{n} - \frac{d_{P_i}}{2} (\vec{\omega}_{P_i} \times \vec{n}) - \frac{d_{P_j}}{2} (\vec{\omega}_{P_j} \times \vec{n}) , \quad (4.13)$$

ergibt sich der tangential zur Partikeloberfläche gerichtete Einheitsvektor \vec{t} zu:

$$\vec{t} = \frac{\vec{v}_{fc}}{|\vec{v}_{fc}|} . \quad (4.14)$$

Damit sind alle Größen definiert, die zur Berechnung der Komponenten des Kraftimpulses benötigt werden. Die Normalkomponente des Kraftimpulses ergibt sich aus:

$$J_n = \frac{(1 + e_p) (\vec{n} \cdot \vec{v}_{rel,c})}{\frac{1}{m_{P_i}} + \frac{1}{m_{P_j}}} , \quad (4.15)$$

wobei e_p die Stoßzahl für den Partikel-Partikel-Stoß ist. Wie beim Partikel-Wand-Stoß unterscheidet man auch hier zwischen Gleit- und Haftstoß. Die Größe der tangentialen Komponente des Kraftimpulses J_t ist abhängig von der Stoßart:

$$1. \text{ Gleitstoß für } \frac{-2 |\vec{v}_{rel,c}|}{7 f_p \left(\frac{1}{m_{P_i}} + \frac{1}{m_{P_j}} \right)} \leq J_n < 0 \quad :$$

$$J_t = -f_p J_n , \quad (4.16)$$

$$2. \text{ Haftstoß für } J_n < \frac{-2 |\vec{v}_{rel,c}|}{7 f_p \left(\frac{1}{m_{P_i}} + \frac{1}{m_{P_j}} \right)} \quad :$$

$$J_t = \frac{2 |\vec{v}_{rel,c}|}{7 \left(\frac{1}{m_{P_i}} + \frac{1}{m_{P_j}} \right)} . \quad (4.17)$$

Hierin ist f_p der Gleitreibungsbeiwert für den Partikel-Partikel-Stoß.

Nachdem der Kraftimpuls \vec{J} auf diese Weise bestimmt worden ist, können mit Hilfe der Gleichungen (4.5)–(4.8) die Geschwindigkeiten und Rotationsgeschwindigkeiten der beiden Partikel nach dem Stoß berechnet werden.

4.4 Modelle für die Simulation der Kollisionen

In diesem Abschnitt sollen die unterschiedlichen Modelle zur Simulation von Partikel-Partikel-Kollisionen vorgestellt werden, die in den vergangenen Jahren von verschiedenen Autoren entwickelt wurden. Man kann dabei grundsätzlich zwei Arten von Modellen unterscheiden. Zum einen existieren Modelle, die auf der sukzessiven Berechnung von Trajektorien beruhen. Diese Art der Modellierung wird im Abschnitt 4.4.1 beschrieben. Die zweite Möglichkeit der Modellierung von Kollisionen basiert auf dem Verfahren der simultanen Partikelverfolgung. Diese Modelle werden im Abschnitt 4.4.2 dargestellt.

4.4.1 Trajektorienbasierte Kollisionsmodelle

Wie im Abschnitt 3.3.2 dargestellt wurde, wird im Verfahren der sukzessiven Trajektorienberechnung eine große Anzahl von Partikeltrajektorien nacheinander berechnet. Nach der Berechnung aller Trajektorien ergeben sich die Feldgrößen der dispersen Phase für ein Kontrollvolumen aus der Mittelwertbildung über alle Trajektorienabschnitte, die in diesem Kontrollvolumen enthalten sind. Dies bedeutet, daß zum Zeitpunkt der Berechnung einer einzelnen Trajektorie diese lokalen Feldgrößen nicht bekannt sind. Andererseits ist leicht einzusehen, daß zur Berechnung von Kollisionen Informationen über die Umgebung des gerade betrachteten Partikels vorliegen müssen. Auf dieser Grundlage wurde von Oesterlé und Petitjean [34] ein iteratives Verfahren entwickelt, das im folgenden Iterierte Trajektorienberechnung (ITB) genannt werden soll.

Das ITB-Verfahren beruht auf einem iterativen Algorithmus, der in der Abbildung 4.2 schematisch dargestellt ist. In einem ersten Iterationsschritt werden die Trajektorien einer repräsentativen Anzahl von Partikeln ohne Berücksichtigung von Inter-Partikel-Kollisionen berechnet. Aus der Berechnung aller Trajektorien ergibt sich die Verteilung des Partikelmassenstroms innerhalb der Strömungsgeometrie für eine stationäre kollisionsfreie Gas-Partikel-Strömung. Des weiteren ergeben sich für jede Gitterzelle des numerischen Gitters mittlere lokale Werte für die Partikelgeschwindigkeit \vec{v}_{P0} , die Rotationsgeschwindigkeit $\vec{\omega}_{P0}$, den Partikeldurchmesser d_{P0} sowie die Anzahldichte n_P der Partikel.

Dieser erste Iterationsschritt des ITB-Algorithmus ist identisch mit dem weitverbreiteten Verfahren zur Lagrangeschen Simulation von stationären, verdünnten dispersen Mehrphasenströmungen, bei dem weder Partikel-Partikel-Kollisionen noch die Rückwirkung der Partikelbewegung auf die fluide Phase berücksichtigt werden.

Im nächsten Iterationsschritt wird die Trajektorienberechnung mit Berücksichtigung der Kollisionen wiederholt. Dabei wird nach jedem Partikelschritt auf der Grundlage

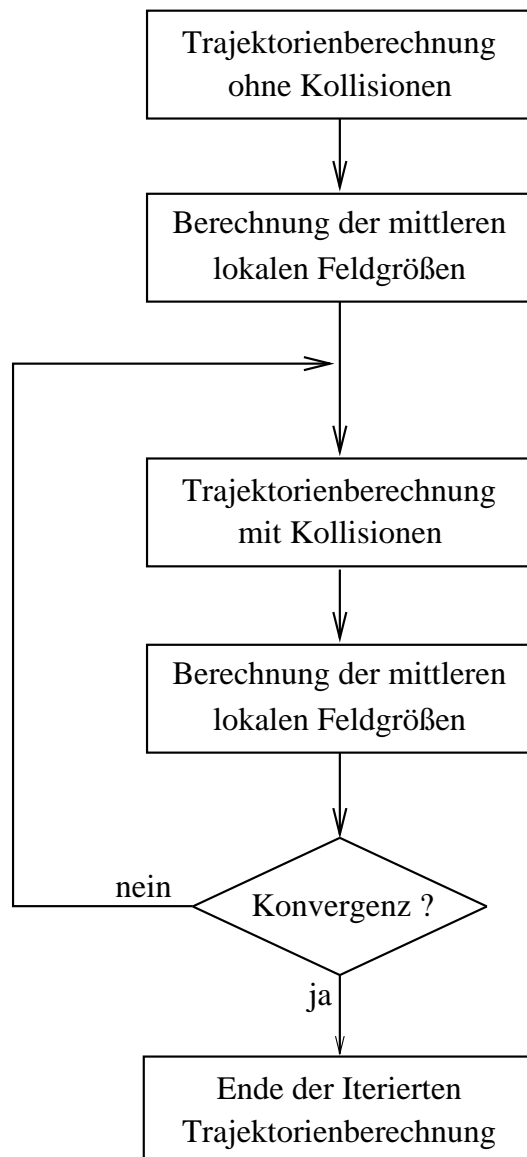


Abbildung 4.2: Algorithmus der Iterierten Trajektorienberechnung (ITB)

der in der vorherigen Iteration berechneten lokalen Feldgrößen mit Hilfe eines stochastischen Verfahrens ermittelt, ob eine Kollision stattfindet. Sind auf diese Weise alle Trajektorien neu berechnet worden, ergibt sich eine neue Verteilung der Partikelkonzentration und der übrigen Feldgrößen. Dieser Vorgang wird solange wiederholt, bis ein konvergenter Zustand eintritt.

Wie bereits oben erwähnt, wird zur Detektion der Kollisionen ein stochastisches Verfahren verwendet. Die Grundlage hierfür bildet die Analogie zwischen einer Gas-Partikel-Strömung und der Strömung eines verdünnten Gases. Die Frequenz der Kollisionen, die ein Partikel mit dem Durchmesser d_P und der Geschwindigkeit \vec{v}_P erfährt, ergibt sich aus der kinetischen Gastheorie [34, 3] zu:

$$\nu_C = \frac{\pi}{4} (d_P + d_{P_0})^2 |\vec{v}_P - \vec{v}_{P_0}| n_P , \quad (4.18)$$

wobei d_{P_0} und \vec{v}_{P_0} der mittlere Durchmesser bzw. die mittlere Geschwindigkeit der umgebenden Partikel sind. Davon ausgehend wird von Oesterlé und Petitjean [34] die Wahrscheinlichkeit P_C für das Stattfinden einer Kollision in einem Zeitintervall Δt hergeleitet:

$$P_C(\Delta t) = 1 - e^{-\nu_C \Delta t} . \quad (4.19)$$

Die Größe der Kollisionswahrscheinlichkeit in Abhängigkeit vom Produkt $\nu_C \Delta t$ ist in der Abbildung 4.3 graphisch dargestellt. Nach jedem Schritt eines Partikels entlang seiner Trajektorie wird auf diese Weise die Wahrscheinlichkeit dafür berechnet, daß während dieses Zeitschrittes eine Kollision stattgefunden hat. Um innerhalb des Simulationsalgorithmus zu entscheiden, ob tatsächlich eine Kollision ausgeführt wird, wird eine gleichverteilte Zufallszahl ψ aus dem Intervall $[0,1]$ generiert. Diese wird mit der aktuellen Kollisionswahrscheinlichkeit verglichen. Falls $P_C \geq \psi$ ist, findet eine Kollision statt, und falls $P_C < \psi$ ist, findet keine Kollision statt.

Wird das Stattfinden einer Kollision detektiert, so erfolgt diese zwischen dem aktuell betrachteten Partikel und einem fiktiven Kollisionspartner. Die physikalischen Eigenschaften des fiktiven Partikels, d.h. sein Durchmesser, seine Geschwindigkeit und Rotationsgeschwindigkeit, sind identisch mit den in der vorherigen Iteration bestimmten mittleren lokalen Größen d_{P_0} , \vec{v}_{P_0} und $\vec{\omega}_{P_0}$. Zur Berechnung des Stoßvorgangs wird die Bewegung des aktuellen Partikels relativ zum fiktiven Partner betrachtet. Die Konfiguration der beiden Partikel zum Stoßzeitpunkt, die in der Abbildung 4.4 dargestellt ist, ergibt sich ebenfalls auf stochastische Weise. Dabei wird der Abstand x_{PC} des aktuellen Partikels von der y -Achse als gleichverteilte Zufallsgröße aus dem Intervall $[-\frac{1}{2}(d_P + d_{P_0}), \frac{1}{2}(d_P + d_{P_0})]$ bestimmt. Die Partikelgeschwindigkeit und -rotationsgeschwindigkeit kann nun mit Hilfe der Impuls- bzw. Drehimpulsgleichungen wie im Abschnitt 4.3 angegeben berechnet werden. Im Verfahren der Iterierten Trajektorienberechnung werden die Größen nach dem Stoß

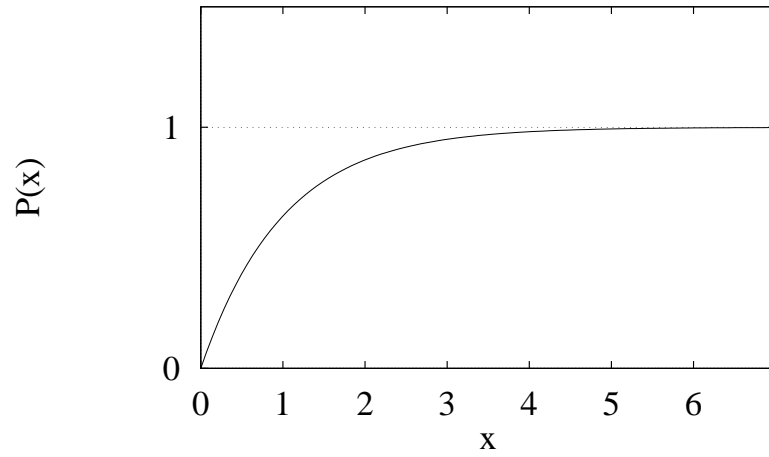


Abbildung 4.3: Abhängigkeit der Kollisionswahrscheinlichkeit von $x = \nu_c \Delta t$

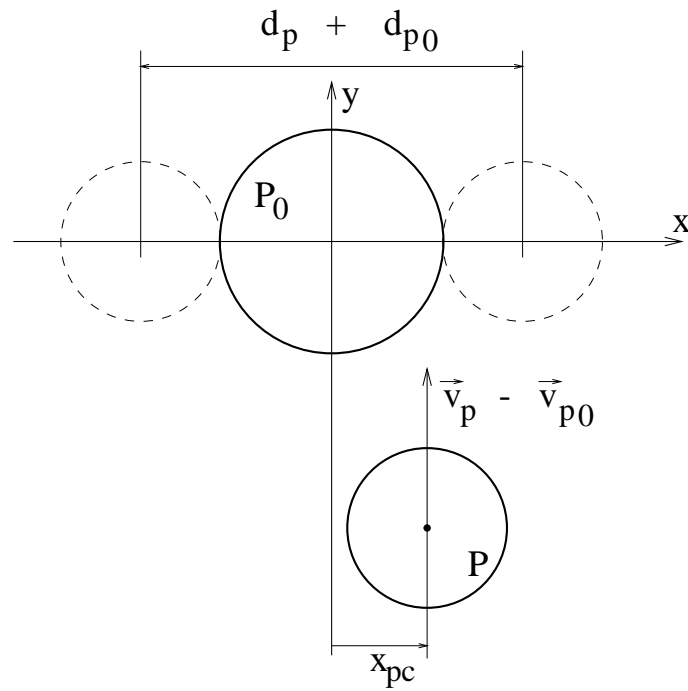


Abbildung 4.4: Konfiguration beim Partikel-Partikel-Stoß

jedoch nur für das aktuell betrachtete Partikel berechnet, während die entsprechenden Größen des fiktiven Kollisionspartners unverändert bleiben.

Ein ähnliches iteratives Verfahren wurde von Sommerfeld [42] vorgeschlagen. Im Unterschied zum oben beschriebenen Verfahren von Oesterlé und Petitjean [34] werden hier die Eigenschaften des fiktiven Partikels nicht aus den mittleren lokalen Feldgrößen bestimmt. Der Partikeldurchmesser und die Geschwindigkeit werden als Zufallswerte aus der lokalen Verteilungsfunktion dieser Größen bestimmt. Diese Verteilungsfunktionen müssen in jeder Iteration für alle Kontrollvolumen berechnet und gespeichert werden, was eine Erhöhung des Rechen- bzw. Speicheraufwandes der Simulation zur Folge hat.

Des weiteren wird in [42] bei der Generierung der Geschwindigkeit des fiktiven Partikels berücksichtigt, daß die Geschwindigkeiten zweier kollidierender Partikel in einem gewissen Maße korreliert sind. Die Ursache für diese Korrelation ist, daß die Partikel sich zum Zeitpunkt der Kollision in demselben turbulenten Fluidwirbel aufhalten. Die Größe der Korrelation ist u.a. abhängig von der aerodynamischen Relaxationszeit τ_A .

Bisher sind erst wenige Anwendungen des ITB-Verfahrens bekannt. Oesterlé und Petitjean [34] verwendeten das Verfahren, um die Strömung von Glaspartikeln und Luft in einem horizontalen Rohr zu berechnen. Ihre Ergebnisse werden qualitativ durch Messungen von Morikawa et al. [32] bestätigt. Ein quantitativer Vergleich mit diesen Experimenten ist nicht möglich, da dort ein größerer Partikel- und Rohrdurchmesser sowie eine niedrigere Gasgeschwindigkeit verwendet wurde. Huber [21] führte vergleichende Simulationen und Experimente zu einer Gas-Partikel-Strömung in einem vertikalen Rohr hinter einem Krümmer durch. Dabei zeigten sich Unterschiede zwischen der Simulation und den experimentellen Ergebnissen, die auf das Ausbilden einer instationären Sträne zurückgeführt wurden, die in einer trajektorienbasierten Berechnung nicht erfaßt werden kann.

Insgesamt läßt sich feststellen, daß bisher keine systematischen Vergleiche dieses Berechnungsverfahrens mit Experimenten bekannt sind. Des weiteren existieren bisher keine Untersuchungen des Verfahrens hinsichtlich seines Konvergenzverhaltens oder der zu verwendenden Konvergenzkriterien. Für solche Untersuchungen wären eine Vielzahl von Simulationen notwendig und sie wären wegen der damit verbundenen langen Rechenzeiten äußerst aufwendig.

Aus diesem Grund ist es ein Ziel dieser Arbeit, durch die Entwicklung paralleler Algorithmen für das ITB-Verfahren die Rechenzeiten signifikant zu verkürzen. Dadurch soll die Voraussetzung dafür geschaffen werden, das Verfahren effizient zu untersuchen und weiterzuentwickeln.

4.4.2 Kollisionsmodelle auf der Basis der simultanen Partikelverfolgung

Deterministische Simulation (DS)

Das im folgenden beschriebene Rechenverfahren, das hier Deterministische Simulation (DS) genannt werden soll, wurde von Tanaka und Tsuji [46] vorgeschlagen. Es basiert auf der Methode der simultanen Partikelverfolgung. Im DS-Verfahren repräsentiert jedes simulierte Partikel genau ein physikalisches Partikel. Die Anzahl der simulierten Partikel ist daher identisch mit der Anzahl der Partikel in der realen Strömung, d.h. $N_{sim} = N_{real}$. Die Bewegung aller Partikel, die sich zu einem bestimmten Zeitpunkt innerhalb der Strömungsgeometrie aufhalten, wird gleichzeitig berechnet. Die Berechnung der Partikelbewegung und die Berechnung der Kollisionen erfolgt entkoppelt voneinander in zwei separaten Schritten. Das Prinzip des Simulationsalgorithmus kann wie folgt zusammengefaßt werden:

1. Zunächst wird die kollisionsfreie Bewegung aller Partikel in einem Zeitintervall Δt berechnet, wie sie sich aufgrund der Bewegungsgleichungen (siehe Gleichungen (3.28) und (3.34)) ergibt. Die Größe des Zeitschrittes Δt wird dabei so gewählt, daß dieser klein gegenüber der mittleren Kollisionszeit τ_c ist.
2. Anschließend wird überprüft, ob während dieses Zeitschrittes Kollisionen stattgefunden haben. Falls eine Kollision zwischen zwei Partikeln detektiert wird, werden mit Hilfe der Impuls- bzw. Drehimpulsgleichungen die Geschwindigkeiten bzw. Rotationsgeschwindigkeiten der beiden Partikel nach dem Stoß berechnet (siehe Gleichungen (4.5)–(4.8)).

Zur Detektion der Kollisionen, die während des Zeitschrittes Δt stattgefunden haben, wird die relative Bewegung jeweils zweier Partikel in Δt betrachtet. Wie in der Abbildung 4.5 dargestellt ist, seien die Partikel mit i und j bezeichnet, und es werde die Bewegung des Partikels j relativ zum Partikel i betrachtet. Bezeichnet man den weiteren den Abstandsvektor zwischen den Partikelmittelpunkten mit \vec{r} , so beträgt der Abstand r_{ij} zwischen den Partikelmittelpunkten zu einem beliebigen Zeitpunkt während des Schrittes Δt :

$$r_{ij} = |\vec{r}_0 + k (\vec{r}_{\Delta t} - \vec{r}_0)|, \quad (4.20)$$

wobei k ein dimensionsloser Parameter mit $0 \leq k \leq 1$ ist. Eine Kollision zwischen den Partikeln findet genau dann statt, wenn ihr Abstand gleich der Summe ihrer Radien ist. Unter Berücksichtigung der Gleichung (4.20) bedeutet dies, daß die Gleichung:

$$|\vec{r}_0 + k (\vec{r}_{\Delta t} - \vec{r}_0)| = \frac{d_{Pi}}{2} + \frac{d_{Pj}}{2}, \quad (4.21)$$

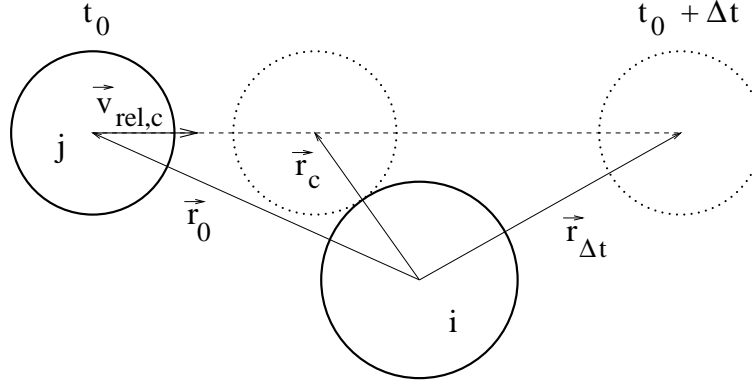


Abbildung 4.5: Detektion einer Kollision zwischen zwei Partikeln

zwei reelle Lösungen k_1 und k_2 mit $k_1 < k_2$ und $0 \leq k_1 < 1$ besitzt. Ist diese Bedingung erfüllt, so ergibt sich der Abstandsvektor \vec{r}_c zum Zeitpunkt der Kollision zu:

$$\vec{r}_c = \vec{r}_0 + k_1 (\vec{r}_{\Delta t} - \vec{r}_0) . \quad (4.22)$$

Damit ist die relative Lage der beiden Partikel zueinander bestimmt. Des weiteren wird angenommen, daß die Translations- und Rotationsgeschwindigkeiten der Partikel vor dem Stoß identisch mit den entsprechenden Größen zum Zeitpunkt $t_0 + \Delta t$, d.h. nach Vollendung des kollisionsfreien Schrittes, sind.

Das Verfahren der Deterministischen Simulation hat den Vorteil, daß die physikalischen Verhältnisse in der realen Gas-Partikel-Strömung sehr gut wiedergegeben werden. Dies liegt zum einen daran, daß die Anzahl der simulierten Partikel identisch mit der Anzahl der realen Partikel in dem betrachteten Strömungsgebiet ist. Zum anderen finden Stöße nur zwischen Partikeln statt, die sich bei ihrer Bewegung tatsächlich berühren, wobei die Konstellation zum Stoßzeitpunkt, d.h. der Berührungspunkt, die Geschwindigkeiten und Rotationsgeschwindigkeiten beim Zusammenprall exakt deterministisch bestimmt werden können.

Der Nachteil dieses Verfahrens besteht darin, daß es aufgrund seiner äußerst hohen Anforderungen an die Computerressourcen nur sehr beschränkt anwendbar ist. Betrachtet man z.B. ein einfaches Rohrleitungssegment mit einem Durchmesser von $D = 250$ mm und einer Länge von $L = 1$ m, in dem sich ein Gemisch von Kohlepartikeln mit einem Durchmesser von $d_p = 50 \mu\text{m}$ und Luft bei einer Massenbeladung von $\eta = 1$ aufhält, so befinden sich in diesem Gebiet ca. $3.6 \cdot 10^8$ Partikel. Nimmt man an, daß zum Speichern der wichtigsten Partikelgrößen wie Koordinaten, Geschwindigkeiten, Rotationsgeschwindigkeiten, usw. ein Speicherplatzbedarf von etwa 100 Byte pro Partikel erforderlich ist, so hätte eine Deterministische Simulation dieses

Rohrsegmentes einen Gesamtspeicherbedarf von ca. 36 000 MB! Zusätzlich dazu ist die Speicherung der Daten für die Fluidströmung erforderlich. Hieran sieht man, daß schon die Simulation einer Strömung mit einer moderaten Massenbeladung in einem relativ kleinen Strömungsgebiet die Speicherkapazität der heute verfügbaren Rechner überschreitet.

Ähnliches gilt für den Rechenaufwand, der zur Detektion der Kollisionen erforderlich ist. Wie oben beschrieben, wird nach jedem Schritt, den die Partikel ausgeführt haben, überprüft, ob während dieses Schrittes eine Berührung zwischen je zwei Partikeln stattgefunden hat. Die Gesamtzahl $N_{P_{aar}}$ der Partikelpaarungen, für die nach jedem Zeitschritt das Stattfinden einer Kollision überprüft werden muß, kann durch die folgende endliche arithmetische Reihe abgeschätzt werden:

$$N_{P_{aar}} \approx (N_{sim} - 1) + (N_{sim} - 2) + \dots + 1 = \frac{1}{2} N_{sim} (N_{sim} - 1), \quad (4.23)$$

wobei N_{sim} die Anzahl der simulierten Partikel ist. In der Praxis ist $N_{P_{aar}}$ etwas kleiner, da der Detektionsalgorithmus für zwei Partikel beendet wird, wenn eine Berührung festgestellt wird. Wie jedoch im Abschnitt 4.2 erläutert wurde, kann angenommen werden, daß während eines Zeitschrittes nur relativ wenige Stöße stattfinden, so daß die Beziehung (4.23) einen guten Schätzwert darstellt. Dies bedeutet, daß nach jedem Zeitschritt die Gleichung (4.21) ca. $N_{P_{aar}}$ -mal gelöst werden muß, wobei $N_{P_{aar}}$ proportional zu N_{sim}^2 ist. Zur Lösung der Gleichung (4.21) müssen pro Partikelpaar mindestens ca. 20 Gleitpunktoperationen ausgeführt werden. Daraus folgt für das oben angegebene Beispiel, daß nach jedem Zeitschritt zur Detektion der Kollisionen ca. $2.6 \cdot 10^{18}$ Gleitpunktoperationen ausgeführt werden müßten. Der schnellste heute verfügbare Rechner bietet eine Rechenleistung von ca. $1.3 \cdot 10^{12}$ Gleitpunktoperationen pro Sekunde [13]. Ein solcher Rechner benötigte daher allein zur Detektion der innerhalb eines Zeitschrittes stattfindenden Kollisionen eine Rechenzeit von ca. 23 Tagen!

Es ist nur eine Arbeit bekannt, in der die Deterministische Simulation angewendet wurde (Tanaka und Tsuji [46]). Hierin wurde die Strömung in einem vertikalen Rohr mit einem Durchmesser von 40 mm berechnet. Der Durchmesser der Partikel betrug $406 \mu\text{m}$ bzw. $1500 \mu\text{m}$. Wegen des geringen Rohrdurchmessers, des relativ großen Partikeldurchmessers sowie der Möglichkeit, am Ein- und Ausströmrand des Strömungsgebietes periodische Randbedingungen zu verwenden, konnte die für die Simulation erforderliche Partikelanzahl auf $N_{sim} = 800$ (für $d_p = 406 \mu\text{m}$) bzw. $N_{sim} = 200$ (für $d_p = 1500 \mu\text{m}$) beschränkt werden. Die Ergebnisse der numerischen Simulation zeigten eine gute Übereinstimmung mit entsprechenden experimentellen Ergebnissen von Tanaka [45].

Aufgrund des übermäßig großen Rechen- und Speicheraufwands ist das Verfahren der Deterministischen Simulation für die Berechnung praxisrelevanter Gas-Partikel-

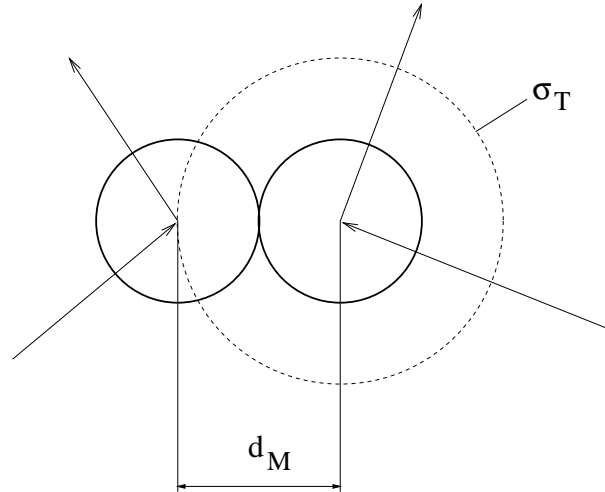


Abbildung 4.6: Kollision von zwei kugelförmigen Molekülen

Strömungen nicht anwendbar. Es ist daher zur Zeit ausschließlich von theoretischem Interesse.

Monte-Carlo-Simulation (MCS)

Um die Nachteile der Deterministischen Simulation zu vermeiden, aber dennoch die Vorteile der simultanen Partikelverfolgung nutzen zu können, wurde von verschiedenen Autoren (z.B. Yonemura et al. [56], Tanaka et al. [47], Kitron et al. [24, 25]) ein Verfahren angewandt, das im folgenden Monte-Carlo-Simulation (MCS) genannt werden soll. Im Unterschied zur DS ist in diesem Verfahren die Anzahl der simulierten Partikel nicht identisch mit der Anzahl der physikalischen Partikel, sondern es wird lediglich eine repräsentative Anzahl von Partikeln simuliert, d.h. $N_{sim} < N_{real}$. Des weiteren erfolgt die Berechnung der Partikel-Partikel-Kollisionen nicht deterministisch, sondern es wird ein stochastischer Algorithmus verwendet. Das Verfahren der Monte-Carlo-Simulation beruht auf der Analogie zwischen der Partikelbewegung in einer Gas-Partikel-Strömung und der Bewegung der Moleküle in der Strömung eines verdünnten Gases. Für die numerische Simulation einer verdünnten Gasströmung wurde von Bird [2, 3] die Direct-Simulation-Monte-Carlo- (DSMC) Methode entwickelt. Diese wurde von den verschiedenen Autoren auf die Simulation kollisionsbehafteter Gas-Partikel-Strömungen übertragen. Aus diesem Grund soll zunächst ein Überblick über die Grundprinzipien der DSMC Methode gegeben werden.

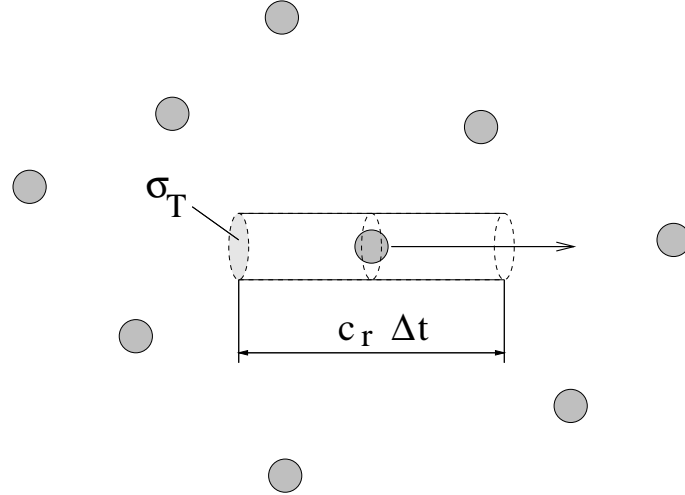


Abbildung 4.7: Effektives Volumen, das relativ zur umgebenden Molekülwolke durchquert wird

Ein verdünntes Gas ist dadurch charakterisiert, daß der mittlere Abstand zwischen den Gasmolekülen δ_M sehr viel größer als deren mittlerer Durchmesser d_M ist:

$$\delta_M \gg d_M. \quad (4.24)$$

Die Moleküle werden hierbei als kugelförmig angesehen. Zwar sind in einem mehratomigen Gas die intermolekularen Kräfte von der Orientierung der Moleküle abhängig, nimmt man jedoch an, daß diese Orientierung zufällig ist und daß die Anzahl der Kollisionen sehr groß ist, so können die molekularen Kraftfelder und damit die Moleküle selbst als kugelförmig betrachtet werden [3]. Für diesen Fall ergibt sich der in der Abbildung 4.6 dargestellte Kollisionsquerschnitt zu:

$$\sigma_T = \pi d_M^2. \quad (4.25)$$

Betrachtet man die Bewegung eines einzelnen Moleküls relativ zu der umgebenden Molekülwolke in einem Zeitintervall Δt , das sehr viel kleiner als die mittlere Kollisionszeit τ_C ist, so beträgt das von diesem Molekül durchquerte effektive Volumen (siehe Abbildung 4.7):

$$\Omega_{eff} = \overline{\sigma_T c_r} \Delta t. \quad (4.26)$$

Hierbei ist c_r die Relativgeschwindigkeit zwischen dem aktuellen und einem benachbarten Molekül, und der Überstrich kennzeichnet eine über alle umgebenden Moleküle gemittelte Größe. Die Anzahl der Moleküle, die sich im Volumen Ω_{eff} aufhalten, ergibt sich mit Hilfe der lokalen Anzahldichte n_M zu:

$$N_\Omega = n_M \overline{\sigma_T c_r} \Delta t, \quad (4.27)$$

woraus für die Anzahl der Kollisionen pro Zeit- und Volumeneinheit, d.h. die Kollisionsfrequenz, folgt:

$$\nu_C = n_M \overline{\sigma_T c_r} . \quad (4.28)$$

Die Simulation einer verdünnten Gasströmung mit der DSMC-Methode beruht nun auf den folgenden Grundprinzipien:

- Simulation einer reduzierten Anzahl von Molekülen, wobei jedes simulierte Molekül eine große Anzahl F_N von realen Molekülen repräsentiert:

$$F_N = \frac{N_{real}}{N_{sim}} . \quad (4.29)$$

Die physikalischen und chemischen Eigenschaften der simulierten Moleküle sind dabei identisch mit denen der realen Moleküle.

- Entkopplung der Molekülbewegung und der intermolekularen Kollisionen. Alle Moleküle werden innerhalb eines Zeitschrittes Δt kollisionsfrei bewegt, wobei $\Delta t \ll \tau_C$ ist. Falls während dieses Schrittes Wechselwirkungen mit den Strömungsberandungen auftreten, z.B. ein Stoß gegen eine Wand, so werden diese Wechselwirkungen berechnet.
- Ausführung einer repräsentativen Anzahl von Kollisionen nach dem Zeitschritt Δt . Diese Anzahl ist im wesentlichen abhängig von der Kollisionsfrequenz und der Größe von Δt .
- Einteilung des Strömungsgebietes in Zellen. Die Größe dieser Zellen muß so gewählt werden, daß die Änderung der makroskopischen Strömungsgrößen in einer Zelle sehr klein ist. Kollisionen finden nur zwischen Molekülen statt, die sich in derselben Zelle befinden, wobei jedoch der Aufenthaltsort der einzelnen Moleküle innerhalb der Zelle dabei nicht berücksichtigt wird.

Die Ausführung der Kollisionen erfolgt mit Hilfe eines stochastischen Verfahrens. Betrachtet man zwei simulierte Moleküle, die sich in einer Zelle befinden und sich mit der Relativgeschwindigkeit c_r zueinander bewegen, so beträgt die Wahrscheinlichkeit für das Stattfinden einer Kollision zwischen den beiden Molekülen während des Zeitintervalls Δt :

$$P_C = \frac{F_N \sigma_T c_r \Delta t}{\Omega_z} , \quad (4.30)$$

wobei Ω_z das Volumen der aktuellen Zelle ist. Eine Möglichkeit zur Berechnung der Kollisionen besteht nun darin, für alle Partikelpaarungen innerhalb einer Zelle die Kollisionswahrscheinlichkeit nach Gleichung (4.30) zu bestimmen und auf dieser Grundlage zu entscheiden, ob zwischen den beiden Partikeln tatsächlich eine

Kollision stattfindet. Der Rechenaufwand, den diese Vorgehensweise erfordert, ist jedoch proportional zum Quadrat der Anzahl der Partikel N_z in der Zelle. Um diesen Rechenaufwand zu verkleinern, wurden verschiedene Verfahren entwickelt, deren Rechenzeitbedarf direkt proportional zu N_z ist.

In der von Bird [2] entwickelten Time-Counter (TC) Methode wird zunächst zufällig ein Molekülpaar ausgewählt. Die Kollisionswahrscheinlichkeit P_C für dieses Molekülpaar wird nach Gleichung (4.30) berechnet. Danach wird eine gleichverteilte Zufallszahl ψ aus dem Intervall $[0,1]$ generiert. Falls $P_C \geq \psi$ ist, findet eine Kollision zwischen diesen beiden Molekülen statt. In diesem Fall wird ein Zeitparameter T_C um einen bestimmten Betrag Δt_C inkrementiert. Dieser Vorgang wird solange wiederholt, bis $T_C \geq \Delta t$ ist.

Die Größe des Zeitinkrements Δt_C kann z.B. mit Hilfe der Kollisionfrequenz bestimmt werden [26]. Unter der Annahme, daß das Zeitintervall zwischen zwei aufeinanderfolgenden Kollisionen Δt_C einer Wahrscheinlichkeitsverteilung unterliegt:

$$P(\Delta t_C) = \nu_C e^{-\nu_C \Delta t_C} , \quad (4.31)$$

kann dieses Zeitintervall als stochastische Größe bestimmt werden aus:

$$\Delta t_C = -\frac{\ln(\psi)}{\nu_C} , \quad (4.32)$$

wobei ψ wiederum eine gleichverteilte Zufallszahl mit $\psi \in [0,1]$ ist. Neben der Tatsache, daß der Rechenaufwand direkt proportional zu N_z ist, besteht ein weiterer Vorteil der TC-Methode darin, daß automatisch die korrekte Kollisionsfrequenz erzielt wird [3].

Ein weiteres Verfahren zur Detektion von Kollisionen besteht in der modifizierten Nanbu-Methode [22]. Hierbei werden die einzelnen simulierten Moleküle in einer Zelle mit einer fortlaufenden Nummer $k = 1, \dots, N_z$ gekennzeichnet. Für jedes Molekül wird nun mit Hilfe einer gleichverteilten Zufallszahl $\psi \in [0,1]$ ein potentieller Kollisionspartner k in folgender Weise ausgewählt:

$$k = [[\psi N_z]] + 1 , \quad (4.33)$$

wobei $[[\psi N_z]]$ der ganzzahlige Anteil des Produkts ψN_z ist. Für die so ausgewählte Molekülpaarung wird mit Hilfe der Gleichung (4.30) die Kollisionswahrscheinlichkeit P_C berechnet. Um zu entscheiden, ob tatsächlich eine Kollision stattfindet, wird die folgende Bedingung ausgewertet:

$$P_C - \frac{k}{N_z} > \psi . \quad (4.34)$$

Falls diese Bedingung erfüllt ist, findet eine Kollision zwischen dem aktuell betrachteten Molekül und dem ausgewählten Kollisionspartner statt. Eine Voraussetzung für die Anwendbarkeit dieses Verfahrens besteht darin, daß für alle möglichen Partikelpaarungen innerhalb einer Zelle gelten muß:

$$P_C < \frac{1}{N_z} . \quad (4.35)$$

Diese Voraussetzung kann durch die Wahl eines ausreichend kleinen Zeitschrittes Δt erfüllt werden. Der Rechenzeitbedarf des modifizierten Nanbu-Verfahrens ist ebenfalls direkt proportional zu N_z . Gegenüber der TC-Methode hat dieses Verfahren den Vorteil, daß zur Detektion eines Stoßes nur eine Zufallszahl generiert werden muß.

Wie bereits zu Beginn dieses Unterabschnittes erwähnt, wurde die Monte-Carlo-Simulation von verschiedenen Autoren für die Berechnung kollisionsbehafteter Gas-Partikel-Strömungen verwendet. Yonemura et al. [56] untersuchten numerisch eine vertikale Kanalströmung für Massenbeladungen von $\eta = 2 - 4$. In diesen Simulationen entwickelte sich durch die wechselseitige Beeinflussung der beiden Phasen und die Bildung von Partikel-Clustern ein instationärer und instabiler Strömungszustand. Die dabei entstehenden Strömungsstrukturen haben ein ähnliches Aussehen wie experimentell beobachtete Strukturen. Ein quantitativer Vergleich der Rechnungen mit experimentellen Daten wurde jedoch in [56] nicht durchgeführt.

Kitron et al. [24] untersuchten mit Hilfe der DSMC-Methode die Wanderosion und die Wärmeübertragung zwischen Partikeln und einer Wand für eine kollisionsbehaftete Gas-Partikel-Strömung. Die numerischen Ergebnisse wurden mit experimentellen Ergebnissen verglichen, wobei sich jedoch nur eine qualitative Übereinstimmung ergab.

Ähnlich wie beim ITB-Verfahren sind auch für das MCS-Verfahren bisher keine Arbeiten bekannt, in denen die Ergebnisse der numerischen Simulation systematisch mit experimentell ermittelten Daten verglichen wurden. Bisher konnte lediglich gezeigt werden, daß das Verfahren in der Lage ist, die relevanten physikalischen Effekte qualitativ zu reproduzieren. Es konnten jedoch noch keine quantitativen Aussagen über die Leistungsfähigkeit dieser Simulationsmethode gemacht werden. Insbesondere sind die möglichen Anwendungsbereiche des Verfahrens, z.B. hinsichtlich der Partikelbeladung und -anzahldichte, zur Zeit nicht klar. Es ist ebenfalls nicht bekannt, ob für das MCS-Verfahren ähnliche Einschränkungen existieren wie für die DSMC-Methode, die nur auf verdünnte Gase und bei Vorliegen eines molekularen Chaos anwendbar ist [3].

Ein wesentlicher Grund für das Fehlen solcher systematischen Untersuchungen ist wiederum der große Rechenaufwand, der mit solchen Untersuchungen verbunden

ist. In den bisher bekannten Arbeiten wurden nur relativ kleine Partikelzahlen bzw. relativ kleine und einfache Strömungsgeometrien verwendet, um den Rechenaufwand zu begrenzen.

Mit Hilfe der im Rahmen dieser Arbeit entwickelten parallelen Algorithmen für die Monte–Carlo–Simulation soll die Voraussetzung dafür geschaffen werden, das MCS–Verfahren effizient zu untersuchen und weiterzuentwickeln. Durch die Verkürzung der Rechenzeiten, die durch den Einsatz massiv paralleler Rechner erreicht werden kann, wird außerdem die Simulation größerer Partikelzahlen und die Untersuchung von Strömungen in komplexeren Geometrien ermöglicht.

Kapitel 5

Algorithmen für die parallele Simulation

In diesem Kapitel sollen die Algorithmen erläutert werden, die die Simulation disperser Mehrphasenströmungen auf massiv parallelen Rechensystemen ermöglichen. Dazu wird zunächst im Abschnitt 5.1 ein kurzer Überblick über die Eigenschaften und das Leistungsvermögen von Parallelrechnern gegeben, und es werden einige wichtige Begriffe erklärt. Zwei wichtige Größen zur Beurteilung der Leistungsfähigkeit eines parallelen Algorithmus, die Effizienz und der Speedup, werden im Abschnitt 5.2 vorgestellt. Anschließend wird im Abschnitt 5.3 das hier verwendete Verfahren zur parallelen Berechnung der Fluidströmung beschrieben. Im Abschnitt 5.4 werden schließlich die Methoden für die parallele Simulation der kollisionsbehafteten Partikelbewegung vorgestellt, die im Rahmen dieser Arbeit neu entwickelt wurden.

5.1 Motivation und Allgemeines

Betrachtet man die Entwicklung der Rechentechnik in den vergangenen Jahrzehnten, so wurden bis etwa zur Mitte der siebziger Jahre Steigerungen der Rechenleistung vor allem durch die Verkleinerung von Schalt- und Taktzeiten der Computer erzielt (siehe Abbildung 5.1) [20]. Nach diesem Zeitpunkt ist jedoch eine deutliche Verlangsamung bzw. eine Stagnation dieser Entwicklung zu erkennen. Dies ist dadurch begründet, daß man zunehmend an physikalische Grenzen stieß, die z.B. durch die Lichtgeschwindigkeit oder die Größe von Molekülen gegeben sind. Betrachtet man im Vergleich dazu die in der Abbildung 5.2 dargestellte Entwicklung der Rechengeschwindigkeit [20], so läßt diese keine derartige Stagnation erkennen. Die nach Mitte der siebziger Jahre erzielten Steigerungen der Rechenleistung sind

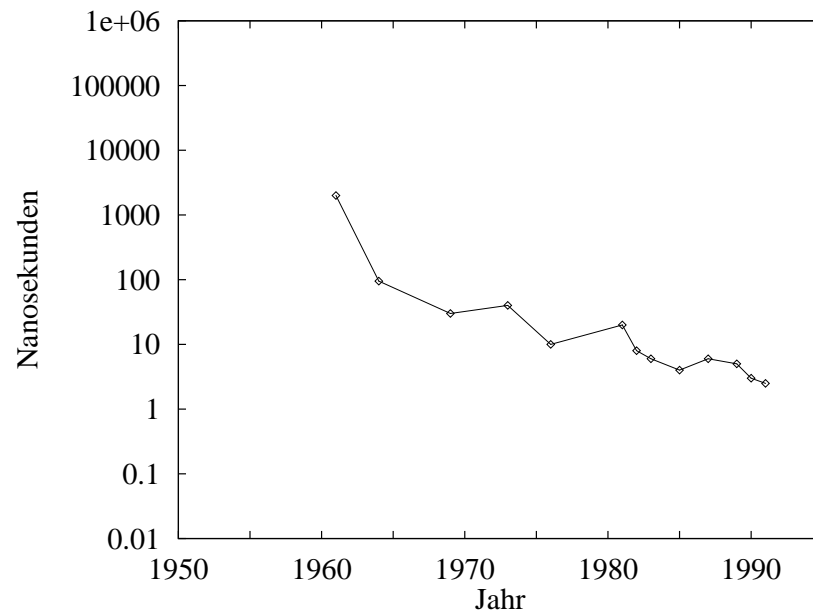


Abbildung 5.1: Entwicklung der Taktzeiten

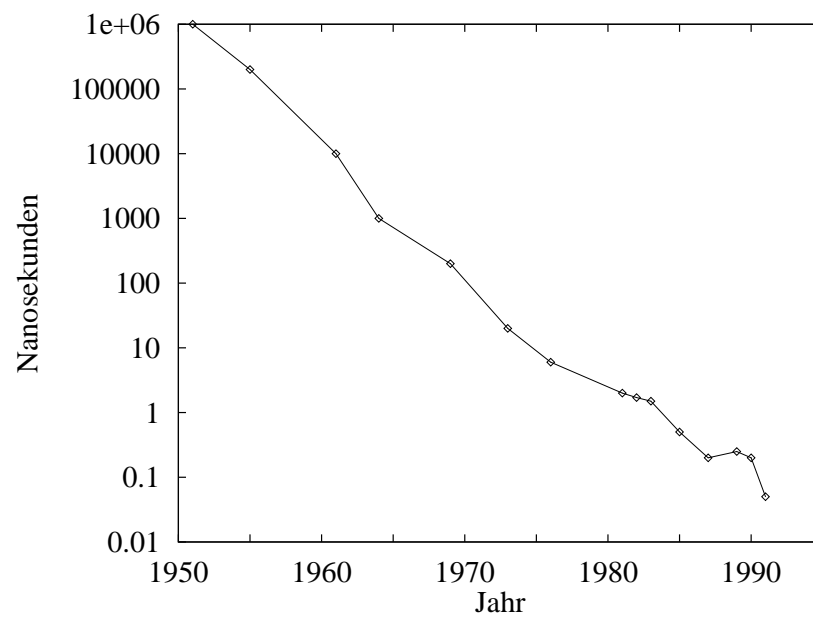


Abbildung 5.2: Entwicklung der Rechengeschwindigkeit (Zeit für die Multiplikation zweier Gleitkommazahlen)

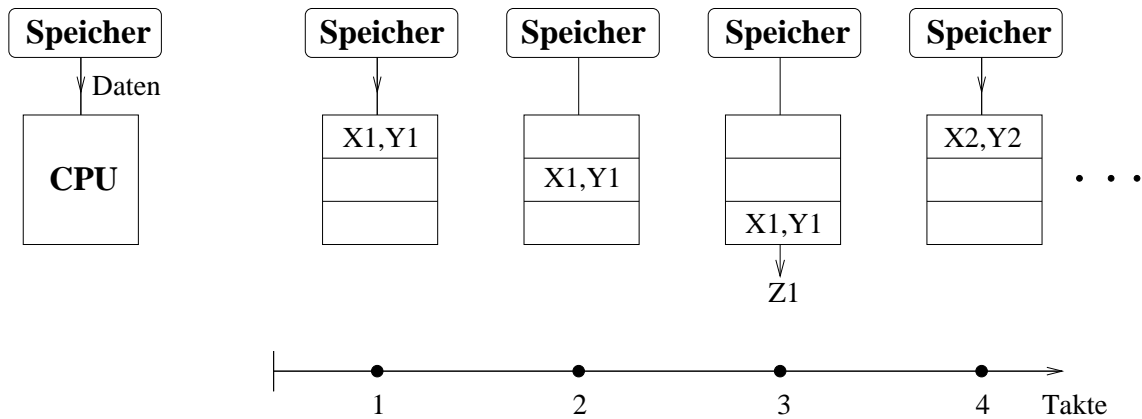


Abbildung 5.3: Arbeitsweise eines skalaren Rechners

fast ausschließlich auf die Einführung verschiedener Arten paralleler Datenverarbeitung zurückzuführen. So können z.B. unterschiedliche Funktionen eines Rechners durch separate Einheiten gleichzeitig wahrgenommen werden. In fast allen Rechnern existieren gesonderte Eingabe-/Ausgabe- (E/A) Einheiten, die simultan zur eigentlichen Datenverarbeitung in der zentralen Recheneinheit (Central Processing Unit – CPU) arbeiten und die E/A-Aufgaben ausführen. Ein Beispiel für Parallelität auf der Ebene der Bitverarbeitung sind parallel arbeitende Schaltelemente, die bei arithmetischen Operationen die gleichzeitige Verknüpfung ganzer Bitreihen der Operanden ermöglichen.

Die wichtigsten Arten der Parallelverarbeitung sind jedoch die in Vektorrechnern realisierte Fließband- oder auch Pipeline-Verarbeitung sowie die Vervielfachung der CPUs, die in Multiprozessorsystemen angewendet wird. In herkömmlichen skalaren Rechnern werden die zu verarbeitenden Daten vom Speicher zur CPU transportiert, wo sie eine bestimmte Anzahl von Verarbeitungsschritten durchlaufen, bis letztlich ein Ergebnis vorliegt. Dieses Arbeitsprinzip ist schematisch in der Abbildung 5.3 dargestellt. In diesem Beispiel wird ein Datenpaar (X, Y) mit drei Operationen in der CPU verarbeitet und das Ergebnis Z erzeugt. Jede Operation findet während eines Taktzyklus statt, so daß nach jeweils drei Taktzyklen ein neues Ergebnis vorliegt.

Das Prinzip der Pipeline-Verarbeitung beruht darauf, daß für Daten, die eine gleichartige Verarbeitung durchlaufen, unterschiedliche Operationen gleichzeitig ausgeführt werden. In dem in Abbildung 5.4 gezeigten Beispiel wird, nachdem die Pipeline gefüllt ist, auf diese Weise in jedem Taktzyklus ein Ergebnis berechnet. Vektorrechner sind vor allem für solche Rechenaufgaben besonders gut geeignet, bei denen eine große Anzahl von Daten gleichartig verarbeitet wird. Dies gilt z.B. für die Mul-

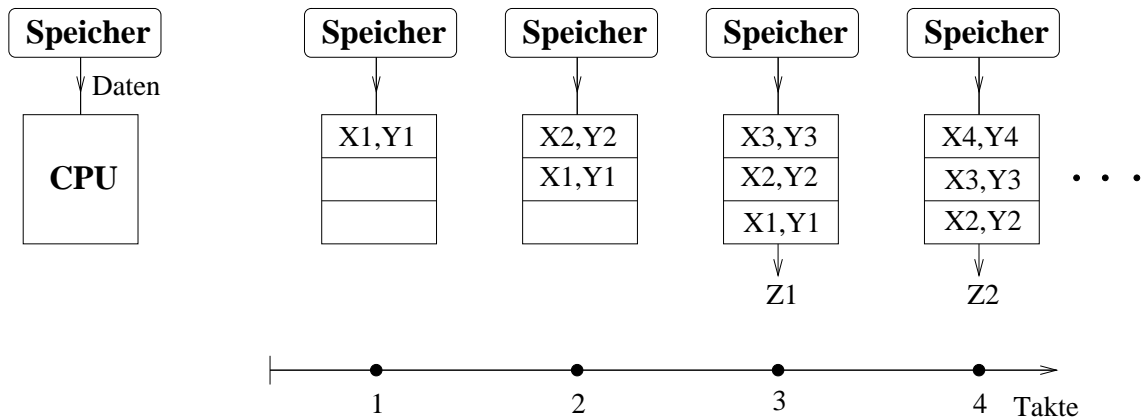


Abbildung 5.4: Arbeitsweise eines Vektorrechners

tiplikation großer Matrizen, die in vielen natur- und ingenieurwissenschaftlichen Anwendungen eine wichtige Rolle spielt.

Multiprozessor-Rechner bestehen aus einer Anzahl von CPUs, die gleichzeitig und voneinander unabhängig die Datenverarbeitung ausführen. Dabei kann die Verarbeitung der Daten in den einzelnen Prozessoren sowohl gleich- als auch verschiedenartig erfolgen. Die Arbeitsweise eines Multiprozessorsystems ist in der Abbildung 5.5 schematisch dargestellt. Hierin werden wiederum zur Berechnung eines Ergebnisses Z drei mathematische Operationen ausgeführt. Diese Berechnung wird in den n Prozessoren des Rechners gleichzeitig mit verschiedenen Eingabedaten durchgeführt, so daß nach jeweils drei Taktzyklen n Ergebnisse produziert werden.

Obwohl, wie oben dargestellt, auf dem Gebiet der Rechentechnik verschiedene Arten der Parallelität realisiert wurden, hat sich die Bezeichnung "Parallelrechner" als Oberbegriff für die Klasse der Multiprozessorsysteme durchgesetzt. Parallelrechner lassen sich hinsichtlich ihrer Architektur unterscheiden in Systeme mit gemeinsamem Speicher (Shared-Memory) sowie Systeme mit verteiltem Speicher (Distributed-Memory). In einem Shared-Memory-Rechner besitzen die Prozessoren einen gemeinsamen Hauptspeicher (siehe Abbildung 5.6). Alle Prozessoren können Daten in diesen Speicher schreiben bzw. aus dem Speicher lesen. Dadurch ist ein Austausch von Informationen zwischen den Prozessoren auf relativ einfache Weise möglich. In einem Distributed-Memory-Rechner hingegen besitzt jeder Prozessor einen eigenen Speicher, auf den die übrigen Prozessoren nicht zugreifen können (siehe Abbildung 5.7). Die einzelnen CPU/Speicher-Einheiten sind untereinander durch ein Netzwerk verbunden. Der Austausch von Informationen zwischen diesen Einheiten erfolgt durch das sogenannte Message-Passing, d.h. das explizite Versenden bzw.

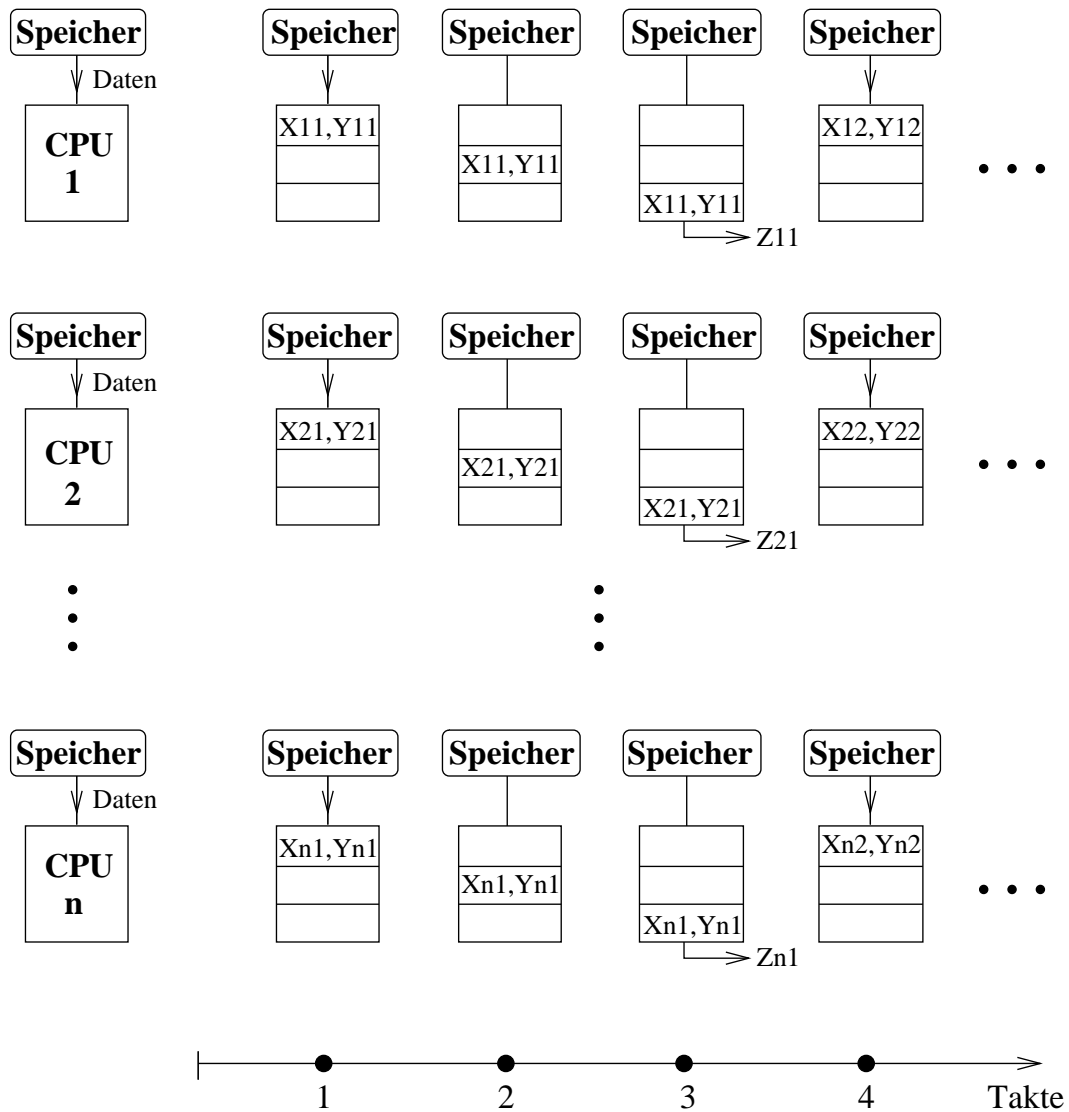


Abbildung 5.5: Arbeitsweise eines Multiprozessor-Rechners

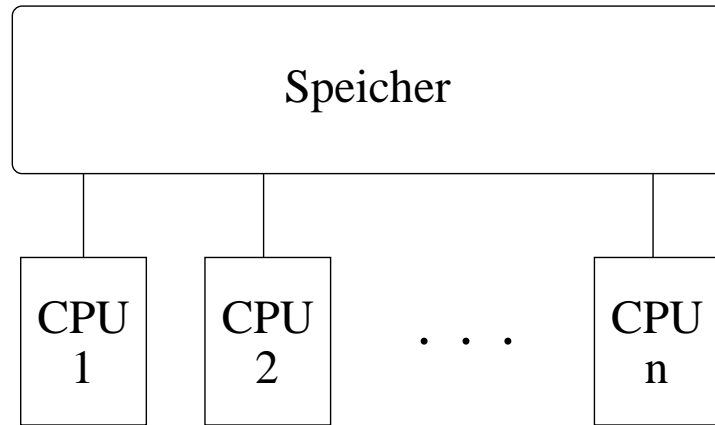


Abbildung 5.6: Prinzip eines Shared-Memory Parallelrechners

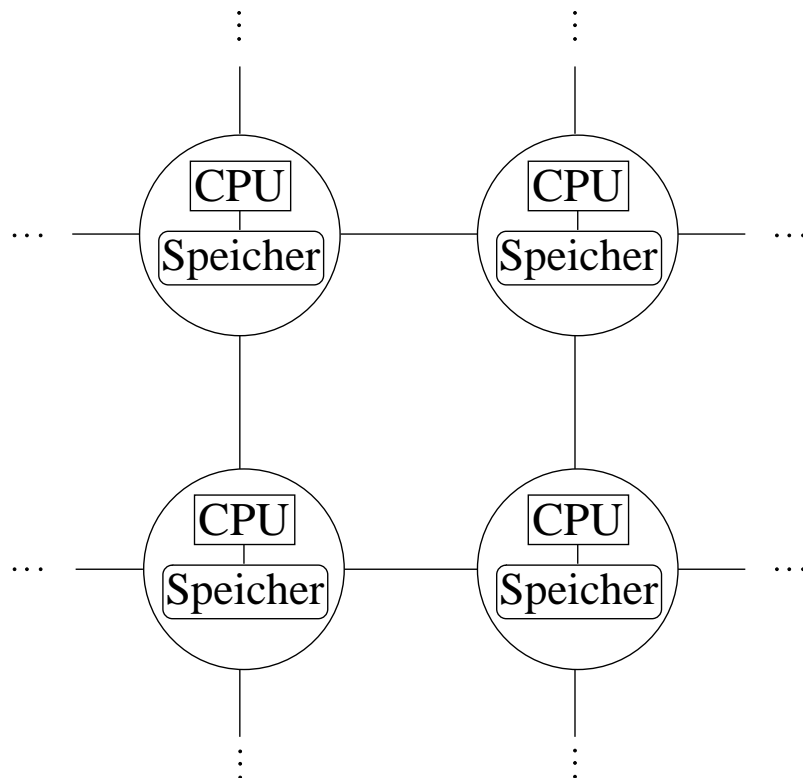


Abbildung 5.7: Prinzip eines Distributed-Memory Parallelrechners

Empfangen von Datenpaketen über das Netzwerk.

Unter den 500 leistungsfähigsten Rechnern, die zur Zeit weltweit installiert sind, befinden sich ausschließlich Multiprozessor-Rechner [13]. Aus technischen Gründen ist die Anzahl von Prozessoren in einem Shared-Memory-Rechner begrenzt. Die derzeit leistungsfähigsten Computer sind daher massiv parallele Systeme mit verteiltem Speicher. Die Anzahl der Prozessoren pro Rechner beträgt dabei bis zu einigen tausend, und die maximale Rechenleistung liegt in der Größenordnung von 10^{12} Gleitpunktoperationen pro Sekunde [13].

Betrachtet man das Gebiet der numerischen Strömungssimulation, so zählt die Lagrangesche Simulation kollisionsbehafteter Gas-Partikel-Strömungen zu den Anwendungen, die die höchsten Anforderungen an die Computerressourcen stellen. Es ist das Ziel dieser Arbeit, das große Leistungspotential der massiv parallelen Rechner für die numerische Berechnung solcher Strömungen zu nutzen. Zu diesem Zweck müssen die verwendeten Simulationsalgorithmen die besondere Architektur der Parallelrechner berücksichtigen. Die in den folgenden Kapiteln vorgestellten Algorithmen verwenden für den Informationsaustausch zwischen den Prozessoren ein explizites Message-Passing, da dies für massiv parallele Rechner mit verteiltem Speicher die einzig mögliche Kommunikationsart ist. Es sei jedoch darauf hingewiesen, daß die Anwendung dieser Algorithmen nicht an eine spezielle Rechnerarchitektur gebunden ist. Für die Implementierung der parallelen Algorithmen wurden standardisierte Message-Passing-Bibliotheken (MPI, PVM) verwendet, so daß sie unabhängig vom speziellen Aufbau der einzelnen Rechner (z.B. hinsichtlich Speicherorganisation, Kommunikationsnetzwerk usw.) allgemein eingesetzt werden können, also auch auf Shared-Memory-Systemen.

5.2 Parameter zur Bewertung paralleler Algorithmen

Es existieren zwei wichtige Parameter, mit deren Hilfe sich die Leistungsfähigkeit eines parallelen Algorithmus beschreiben läßt. Der Speedup Sp ist das Verhältnis der Zeit T_1 , die zur seriellen Berechnung eines Problems benötigt wird, zur Zeit T_N , die N Prozessoren zur Lösung dieses Problems benötigen:

$$Sp(N) = \frac{T_1}{T_N}. \quad (5.1)$$

Falls sich das zu berechnende Problem ideal skalieren läßt und der parallele Algorithmus ideal arbeitet, ist die Größe des Speedups gleich der Anzahl der Prozessoren. Dieser Fall ist im linken Teil der Abbildung 5.8 graphisch dargestellt.

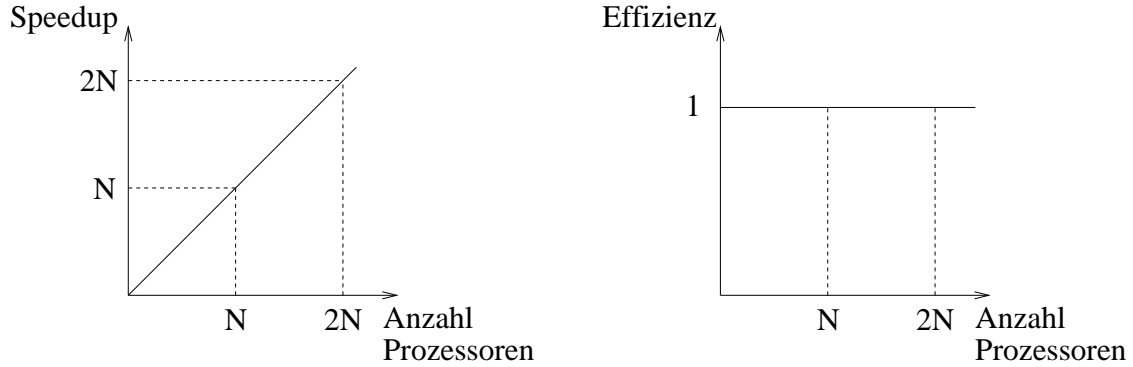


Abbildung 5.8: Idealer Speedup (links) und ideale Effizienz (rechts) in Abhängigkeit von der Anzahl der Prozessoren

Die Effizienz E einer parallelen Berechnung ist definiert als das Verhältnis des Speedups zur Anzahl der für die Berechnung eingesetzten Prozessoren:

$$E(N) = \frac{Sp(N)}{N} = \frac{T_1}{N \cdot T_N} . \quad (5.2)$$

Im Idealfall ist also die Effizienz gleich 1, wie im rechten Teil der Abbildung 5.8 gezeigt.

5.3 Parallele Berechnung der Gasströmung

Die Parallelisierung der Berechnung der Fluidströmung basiert auf der Methode der Gebietszerlegung (Domain Decomposition). Hierbei wird das Strömungsgebiet in kleinere Teilgebiete (Partitionen) aufgeteilt. Die Fluidströmung in den einzelnen Teilgebieten wird von separaten Prozessoren des Parallelrechners gleichzeitig berechnet. Die Partitionierung des numerischen Gitternetzes ist in der Abbildung 5.9 dargestellt. Für die Berechnung der Strömungsgrößen in den Gitterzellen, die unmittelbar an der Grenze zwischen den neu entstandenen Partitionen liegen, ist es erforderlich, daß die Partitionen sich überlappen. Die Strömungsgrößen für ein beliebiges Kontrollvolumen ergeben sich, wie im Kapitel 2 dargestellt, aus den Kräften, die auf dieses Volumen einwirken und den Flüssen durch die Oberfläche des Volumens. Betrachtet man eine Gitterzelle im Inneren eines numerischen Gitters, so ergeben sich diese Kräfte und Flüsse aus den Strömungsgrößen in den umliegenden Gitterzellen (siehe Abbildung 5.10, links). Falls eine Seitenfläche der Zelle zu einem physikalischen Rand (Ein-/Ausströmrand, feste Wand, ...) des Strömungsgebietes

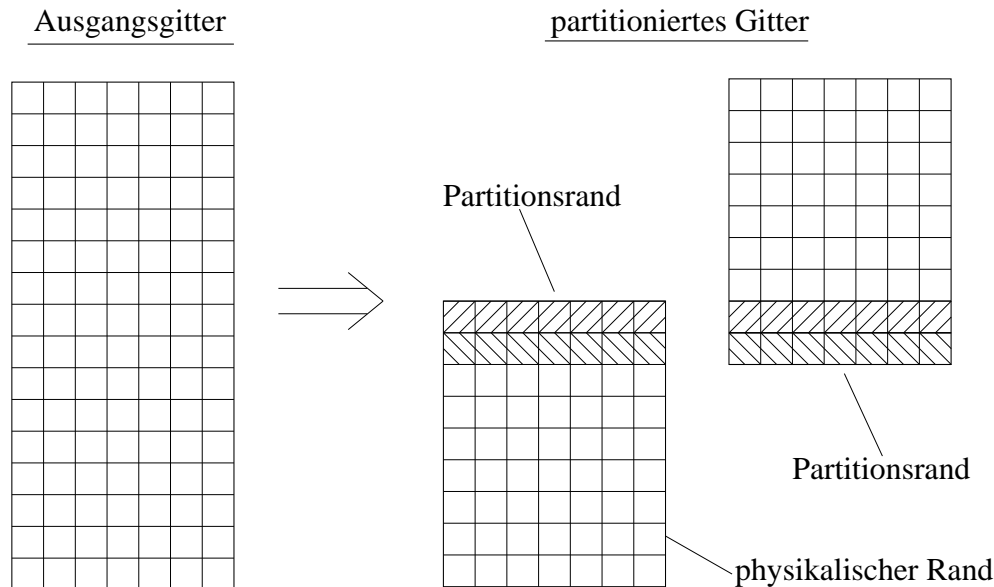


Abbildung 5.9: Prinzip der Gitterpartitionierung

gehört, so werden die Kräfte auf bzw. die Flüsse durch diese Fläche durch die entsprechende physikalische Randbedingung bestimmt (siehe Abbildung 5.10, rechts). Für die Gitterzellen an den Partitionsrändern existiert jedoch keine explizite Randbedingung, da sich diese Zellen im Inneren des Strömungsgebietes befinden. Deshalb wird die Gitteraufteilung so durchgeführt, daß sich die entstehenden Teilgitter in der in den Abbildungen 5.9 bzw. 5.11 gezeigten Weise überlappen. Unter der Annahme, daß die Strömungsgrößen in der randnächsten Gitterzelle bekannt sind, können die Fluidvariablen für die nächste, weiter innen gelegene Zelle auf die oben beschriebene Weise berechnet werden. Nachdem die Fluidgrößen für alle Zellen im Inneren und an den physikalischen Rändern bestimmt worden sind, wird an den Partitions Grenzen ein Datenaustausch durchgeführt. Aufgrund der Überlappung entspricht jeweils die – von außen gesehen – zweite Gitterzelle einer Partition der ersten Gitterzelle der Nachbarpartition (siehe Abbildung 5.11). Die unbekannten Werte für die Strömungsgrößen der Randzelle können daher von der Nachbarpartition übertragen werden, wo sie zuvor explizit berechnet wurden.

Für den Austausch der Randdaten zwischen den Prozessoren, die die verschiedenen Teilgebiete bearbeiten, wird eine gewisse Zeit benötigt, während der die Prozessoren die Strömungsberechnung unterbrechen müssen. Dies bedeutet einen zusätzlichen Zeitaufwand, der zu der eigentlichen Rechenzeit, d.h. der Zeit für die Ausführung mathematischer Operationen, hinzukommt. Um diesen Zusatzaufwand zu minimie-

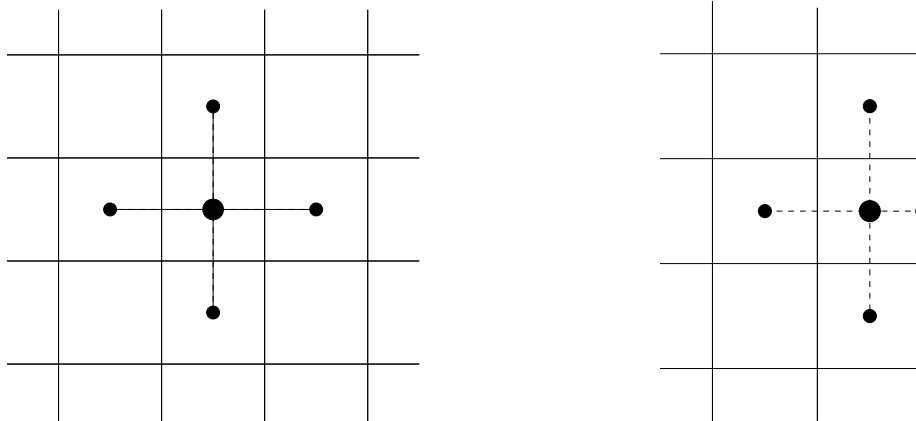


Abbildung 5.10: Gitterzellen im Inneren einer Partition (links) und an physikalischen Rändern (rechts)

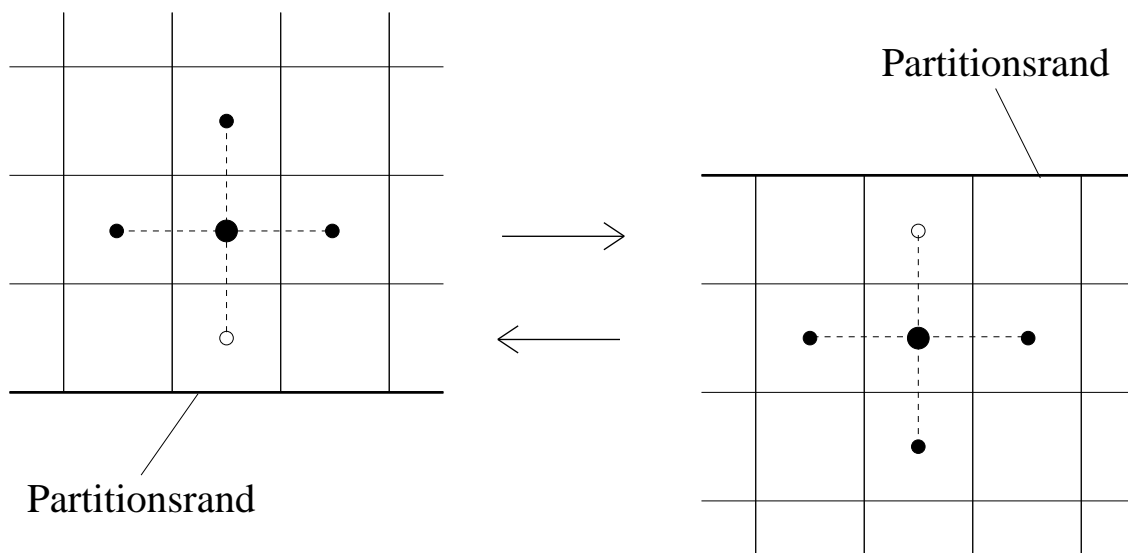


Abbildung 5.11: Gitterzellen an Partitionsrändern

ren, muß die Menge der zwischen den Partitionen auszutauschenden Daten möglichst klein sein. Die Gitteraufteilung sollte daher so durchgeführt werden, daß die Anzahl der Gitterzellen an den Partitionsrändern möglichst klein ist.

Bei dem hier verwendeten Rechenverfahren für die fluide Phase wird zur Berechnung der Strömungsgrößen für jede Zelle des numerischen Gitters derselbe Rechenaufwand benötigt. Aus diesem Grund ist der Rechenaufwand, der in einer Gitterpartition anfällt, proportional zur Anzahl der Gitterzellen in dieser Partition. Hieraus ergibt sich eine weitere Bedingung für eine optimale Gitterpartitionierung: Die resultierenden Teilgitter sollten möglichst gleich groß sein, d.h. die gleiche Anzahl Gitterzellen besitzen. Andernfalls kommt es zu einer ungleichmäßigen Verteilung der Rechenlast unter den Prozessoren des Parallelrechners. Dies führt dazu, daß für Prozessoren mit geringerer Rechenlast Leerlaufzeiten entstehen, während sie auf den Datenaustausch mit Prozessoren mit höherer Rechenlast warten.

5.4 Parallele Berechnung der kollisionsbehafteten Partikelbewegung

In diesem Abschnitt sollen die im Rahmen dieser Arbeit neu entwickelten parallelen Algorithmen für die Simulation der kollisionsbehafteten Partikelbewegung vorgestellt werden. Zunächst wird im Unterabschnitt 5.4.1 eine kurze Zusammenfassung der bisherigen Arbeiten auf diesem Gebiet gegeben. In den Unterabschnitten 5.4.2 und 5.4.3 erfolgt die eigentliche Beschreibung der neuen Algorithmen. Dabei handelt es sich zum einen um Algorithmen für trajektorienbasierte Rechenverfahren und zum anderen um Algorithmen für die Berechnung mit Hilfe der simultanen Partikelverfolgung.

Es sei an dieser Stelle bereits darauf hingewiesen, daß bei der Beschreibung der Algorithmen zwischen den Begriffen “Prozessor“ und “Prozeß“ unterschieden wird. Bei einem “Prozessor“ handelt es sich um eine CPU, d.h. eine Hardware-Komponente eines Rechners. Dagegen ist ein “Prozeß“ eine Software-Komponente, die auf einem Prozessor ausgeführt wird. Viele Rechner bieten die Möglichkeit, gleichzeitig mehrere Prozesse auf einem Prozessor auszuführen.

5.4.1 Stand der Forschung

Auf dem Gebiet der parallelen Berechnung disperser Mehrphasenströmungen mit Hilfe des Lagrange-Verfahrens sind bislang erst wenige Arbeiten bekannt. Tysinger

und Missaghi [52] verwendeten ein paralleles Verfahren zur Simulation einer Gas-Tropfen-Strömung im Brennraum von Verbrennungskraftmaschinen. Zur Berechnung der Gasströmung wurde dabei ein Gebietszerlegungsverfahren verwendet. Die Berechnung der Tropfenbewegung erfolgte mit Hilfe der Trajektorienmethode und unter Anwendung eines Shared-Memory-Ansatzes. Dazu wurde zunächst das Gasgeschwindigkeitsfeld berechnet und dieses in dem gemeinsamen Speicher abgelegt. Anschließend berechnete jeder Prozessor einen bestimmten Anteil der Trajektorien jeweils von ihrem Start- bis zu ihrem Endpunkt im Strömungsgebiet. Ein Vorteil dieses Verfahrens ist, daß durch eine gleichmäßige Aufteilung der Anzahl der zu berechnenden Trajektorien unter den Prozessoren relativ einfach eine gleichmäßige Verteilung der Rechenlast erzielt werden kann. Von Nachteil ist jedoch, daß das Verfahren nur auf Computern mit einer Shared-Memory-Architektur eingesetzt werden kann.

Tsuji [51] berechnete mit Hilfe der simultanen Partikelverfolgung den Einfluß von Partikel-Partikel-Kollisionen in einer zwei-dimensionalen Kanalströmung. Aus persönlichen Mitteilungen von Tsuji ist bekannt, daß für diese Simulationen ein Workstation-Cluster mit einer geringen Anzahl von Workstations verwendet wurde. Über Details des dabei eingesetzten parallelen Verfahrens liegen keine näheren Informationen vor.

In den vergangenen Jahren wurden von Frank [16] bzw. Frank und Wassen [17, 18] parallele Algorithmen für die trajektorienbasierte Simulation entwickelt, die für den Einsatz auf massiv parallelen Rechnern geeignet sind. Diese Verfahren waren jedoch bisher auf die Simulation verdünnter, d.h. kollisionsfreier Gas-Partikel-Strömungen beschränkt..

Insgesamt läßt sich sagen, daß bisher keine Arbeiten bekannt sind, in denen die numerische Simulation kollisionsbehafteter disperser Mehrphasenströmungen auf massiv parallelen Hochleistungsrechnern durchgeführt wurde.

5.4.2 Neue parallele Algorithmen für die trajektorienbasierte Simulation

Zur parallelen Berechnung von Partikeltrajektorien in verdünnten Gas-Feststoff-Strömungen wurden, wie schon oben erwähnt, in den letzten Jahren von Frank [16] bzw. Frank und Wassen [17, 18] zwei verschiedene Verfahren entwickelt, das Domain-Decomposition-Verfahren und das Distributed-Shared-Memory-Verfahren. Im Rahmen dieser Arbeit wurden diese beiden Verfahren im Hinblick auf die Einbeziehung der Partikel-Partikel-Kollisionen weiterentwickelt.

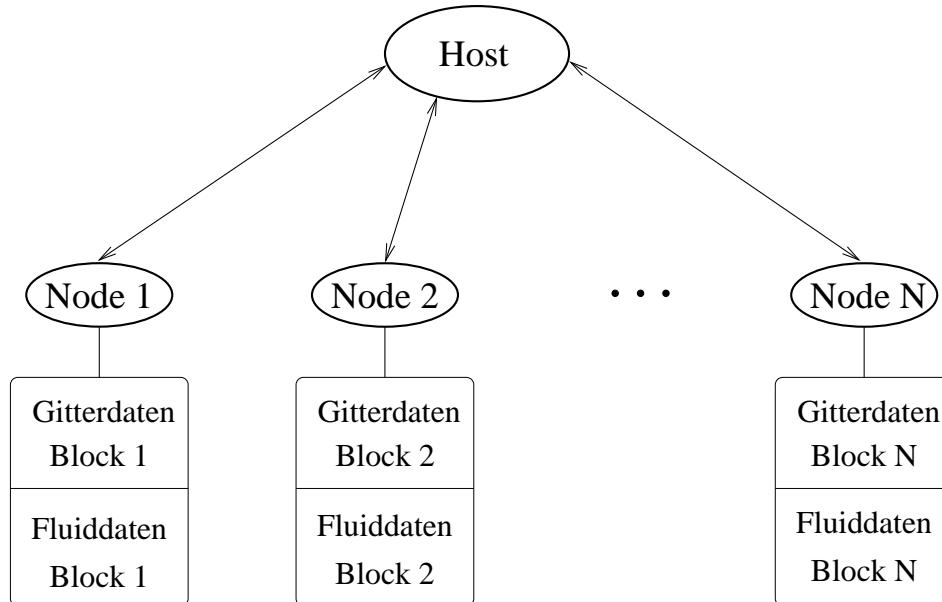


Abbildung 5.12: Prinzip des Domain-Decomposition-Verfahrens

Das Domain-Decomposition- (DD) Verfahren

Das Prinzip des Domain-Decomposition-Verfahrens beruht auf dem gleichnamigen Verfahren, das im Abschnitt 5.3 zur parallelen Berechnung der Fluidströmung beschrieben wurde. Das Strömungsgebiet wird in Teilgebiete aufgeteilt, wobei dieselbe Partitionierung verwendet wird, die zuvor auch für die Strömungsberechnung verwendet wurde. Jedes Teilgebiet wird einem separaten Prozeß zugeordnet. Dieser Prozeß speichert lokal und statisch, d.h. während der gesamten Rechnung, sämtliche zu diesem Teilgebiet gehörenden Daten der Gittergeometrie und der Strömungsvariablen. Wie in der Abbildung 5.12 dargestellt ist, existieren im DD-Verfahren zwei Arten von Prozessen: Ein Host-Prozeß und N Node-Prozesse. Die Anzahl N der Node-Prozesse ist dabei gleich der Anzahl der Gitterpartitionen. Die Node-Prozesse übernehmen die Speicherung der Gitter- und Fluiddaten. Des weiteren führen die Node-Prozesse die Berechnung der Partikeltrajektorien durch. Alle Trajektoriensegmente, die einen bestimmten Gitterblock durchqueren, werden von dem Prozeß berechnet, dem dieser Block zugeordnet ist. Der Host-Prozeß führt alle E/A-Vorgänge durch und koordiniert die Trajektorienberechnung der Node-Prozesse.

Die Aufgaben der verschiedenen Prozesse seien im folgenden anhand des in der Abbildung 5.13 dargestellten Beispiels erläutert. Hierbei handelt es sich um eine Strömung entlang einer rückwärts springenden Stufe, für die beispielhaft der Ver-

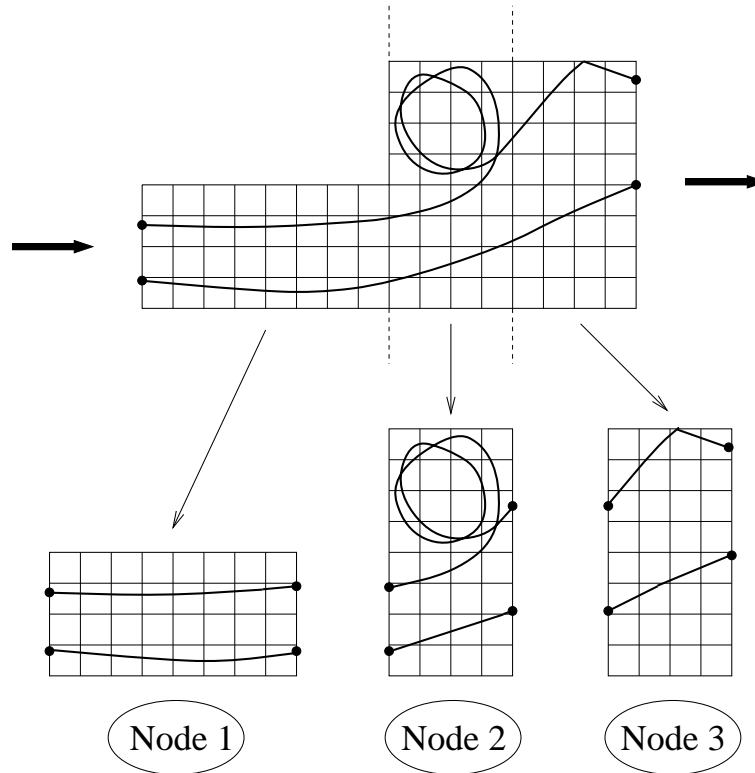


Abbildung 5.13: Beispiel für die Trajektorienberechnung im Domain-Decomposition-Verfahren

lauf von zwei Partikeltrajektorien dargestellt ist. Das Strömungsgebiet wird in drei gleich große Partitionen aufgeteilt, von denen jede einem Node-Prozeß zugeordnet wird. Zu Beginn der Rechnung sind alle Trajektorien-Startbedingungen im Host-Prozeß gespeichert. Eine Startbedingung ist dabei eine Menge von Daten (Koordinaten, Geschwindigkeit, Rotationsgeschwindigkeit, Partikeldurchmesser, ...), die den Startzustand einer Trajektorie bestimmen. In dem in der Abbildung 5.13 gezeigten Beispiel sind zu Beginn der Rechnung alle Startbedingungen am linken Rand der linken Partition lokalisiert. Der Host-Prozeß sendet nun unter Berücksichtigung der Startkoordinaten die Startbedingungen an die Node-Prozesse. Dabei wird eine Startbedingung nur an einen solchen Node-Prozeß gesendet, in dessen Teilgebiet die entsprechende Startbedingung lokalisiert ist. Der Node-Prozeß empfängt die Startbedingung und berechnet das zugehörige Trajektoriensegment in dem ihm zugeordneten Teilgebiet. Wenn eine Abbruchbedingung (z.B. Ausströmrand, Partitionsrand, ...) erreicht wird, wird der Endzustand des Trajektoriensegmentes an den Host zurückgesendet. Falls die Trajektorie einen Partitionsrand erreicht hat, wird

der Endzustand als neue Startbedingung behandelt, die vom Host an denjenigen Node gesendet wird, dem die entsprechende Nachbarpartition zugeordnet ist. Der Austausch von Daten findet in diesem Verfahren ausschließlich zwischen Host und Nodes statt. Die Node-Prozesse tauschen untereinander keine Daten aus.

Durch die Einbeziehung von Partikel-Partikel-Kollisionen ändert sich das Prinzip dieses Algorithmus nicht. Allerdings erhöht sich der Rechenaufwand pro Partikelschritt, da jeweils das Stattfinden einer Kollision überprüft und gegebenenfalls die Kollision ausgeführt werden muß. In Gebieten, in denen sehr viele Kollisionen stattfinden, erhöht sich der Rechenaufwand stärker als in Gebieten mit wenigen Kollisionen. Des weiteren unterscheidet sich im allgemeinen die Verteilung der Trajektorien im Strömungsgebiet im kollisionsbehafteten Fall von der Verteilung im kollisionsfreien Fall. Dies hat zur Folge, daß sich die Verteilung des Rechenaufwandes unter den Gitterpartitionen und damit der Rechenlast unter den Prozessoren unterscheiden kann. Die Auswirkungen, die diese Effekte auf die Leistungsfähigkeit des DD-Verfahrens haben, werden im Kapitel 6 genauer untersucht.

Neben der Erhöhung des Rechenaufwandes hat die Berücksichtigung der Kollisionen außerdem eine Erhöhung des Speicheraufwandes zur Folge. Betrachtet man zunächst die Berechnung der Gasströmung, so müssen im zweidimensionalen Fall für jede Gitterzelle mindestens sieben Variablen gespeichert werden $(x, y, u_F, v_F, p, k, \varepsilon)$. Bei der Simulation einer kollisionsfreien Gas-Partikel-Strömung hängt die Anzahl der zu speichernden Partikelvariablen davon ab, welches Ziel die Simulation hat. Um allgemeine Informationen über die Partikelphase zu erhalten, sei die Speicherung der sechs Variablen $u_{P0}, v_{P0}, \omega_{Px0}, \omega_{Py0}, d_{P0}$ und n_P erforderlich. In diesem Fall wäre zur Simulation einer kollisionsfreien Strömung die Speicherung von insgesamt 13 Variablen pro Gitterzelle erforderlich. Betrachtet man nun die Berechnung einer kollisionsbehafteten Strömung mit Hilfe des ITB-Verfahrens, so ist es notwendig, die oben angegebenen Partikelvariablen nicht nur für die aktuelle Iteration zu speichern, sondern ebenfalls die entsprechenden Größen aus der vorhergehenden Iteration. Durch diese sechs zusätzlich zu speichernden Variablen erhöht sich der Gesamtspeicheraufwand für einen Node-Prozeß im Vergleich zur kollisionsfreien Berechnung um ca. 50%. Dies kann von Nachteil sein, da die Prozessoren vieler massiv paralleler Rechner einen relativ kleinen Hauptspeicher besitzen. Dadurch kann es erforderlich sein, das numerische Gitter in mehr und damit kleinere Partitionen aufzuteilen. Durch die größere Anzahl von Partitionen würden für eine kollisionsbehaftete Berechnung mehr Prozessoren benötigt als für eine kollisionsfreie.

Ein Vorteil des DD-Verfahrens ist seine relativ einfache Implementierung, da es dieselbe Verteilung von Gitter- und Fluiddaten unter den Prozessen verwendet wie der parallele Algorithmus zur Berechnung der Fluidströmung. Ein weiterer Vorteil ist, daß durch die Verwendung des Host-Prozesses als Koordinator die Trajektorienberechnung der Nodes asynchron durchgeführt werden kann. Wenn die Berechnung

eines Trajektoriensegmentes beendet ist, so wird der Endzustand an den Host gesendet, und es kann sofort mit der Berechnung des nächsten Segmentes begonnen werden. Im Falle einer direkten Node–Node–Kommunikation träten häufig Wartezustände auf, da die Zeit, die für die Berechnung eines Segmentes benötigt wird, stark variieren kann.

Ein Nachteil dieses Verfahrens ist, daß die Verteilung der Rechenlast unter den Prozessoren des Parallelrechners sehr ungünstig sein kann [16, 18, 55]. Dies kann verschiedene Ursachen haben. Betrachtet man das Beispiel der rückwärts springenden Stufe in der Abbildung 5.13, so entsteht hinter der Stufe ein Rezirkulationsgebiet, in dem die Partikel längere Zeit verweilen können, falls sie in dieses Gebiet eintreten. Dies hat zur Folge, daß der Rechenaufwand, der in diesem Bereich des Strömungsgebietes anfällt, wesentlich größer ist als in Gebieten, die von den Partikeln schnell durchquert werden. Weitere mögliche Ursachen für eine inhomogene Verteilung des Rechenaufwandes im Strömungsgebiet sind:

- Inhomogene Partikelverteilung im Strömungsgebiet, z.B. in stark separierten Strömungen oder in Freistahlströmungen hinter Düsen.
- Partikel–Wand–Kollisionen, aufgrund der zusätzlich erforderlichen Rechenoperationen zur Detektierung und Berechnung eines Wandstoßes.
- Strömungsgebiete mit großen Gradienten in der Fluidströmung und kleinen turbulenten Zeitmaßen, da in diesen Gebieten aus Gründen der Rechengenauigkeit der Integrationszeitschritt stark verkleinert wird.
- Partikel–Partikel–Kollisionen, da z.B. in Gebieten mit hoher Partikelkonzentration mehr Kollisionen stattfinden als in anderen Gebieten.

Da im DD–Verfahren die Zuordnung der Gitterpartitionen zu den Prozessen statisch ist, hat eine inhomogene Verteilung des Berechnungsaufwandes stets eine inhomogene Verteilung der Rechenlast unter den Node–Prozessen zur Folge. Die Gesamt-rechenzeit für eine parallele Simulation ist immer identisch mit der Rechenzeit, die der Node–Prozeß mit der größten Rechenlast benötigt. Zum Erzielen einer optimalen, d.h. möglichst kurzen, Rechenzeit ist daher eine möglichst gleichmäßige Aufteilung der Rechenlast auf die Node–Prozesse erforderlich. Aus den oben angegebenen Gründen ist bei Anwendung des DD–Verfahrens eine solche optimale Rechenlastverteilung im allgemeinen nicht möglich.

Das Distributed–Shared–Memory– (DSM) Verfahren

Die Nachteile des Domain–Decomposition–Verfahrens können vermieden werden, wenn die Zuordnung von Gitterpartitionen zu Node–Prozessen nicht statisch, son-

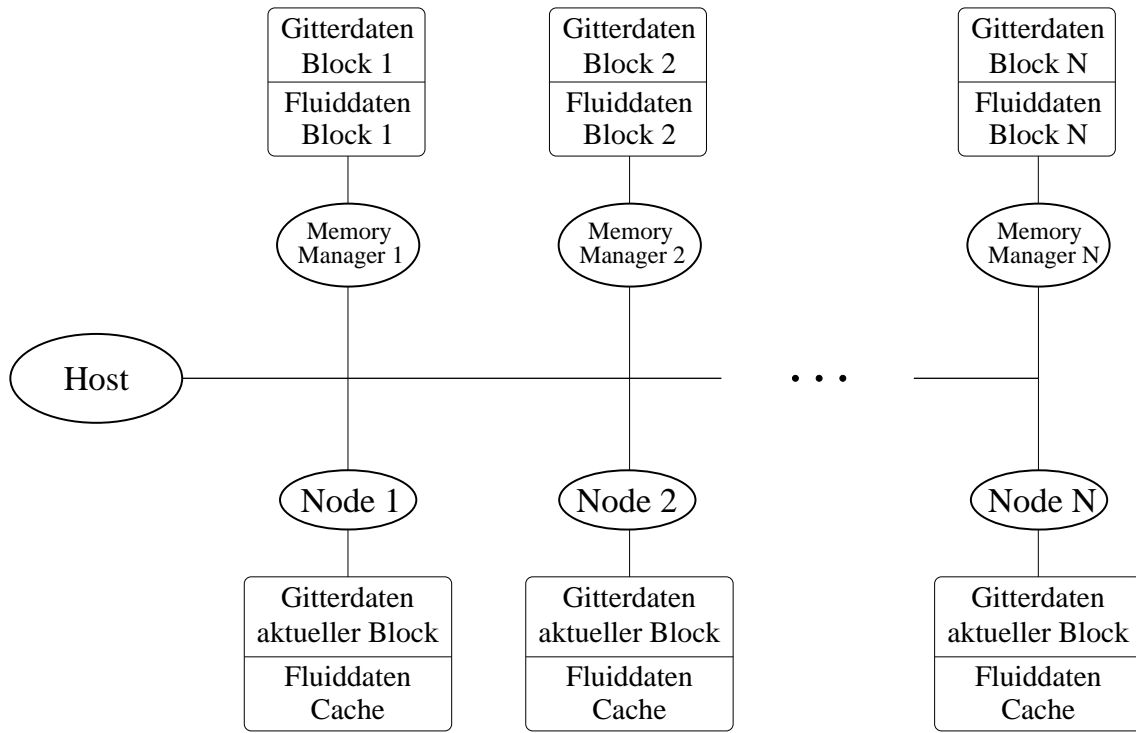


Abbildung 5.14: Prinzip des Distributed-Shared-Memory-Verfahrens

dern dynamisch während der Laufzeit der Rechnung erfolgt. Dies geschieht im Distributed-Shared-Memory-Verfahren. Im Vergleich zum DD-Verfahren wird hier eine zusätzliche Klasse von Prozessen eingeführt, die Memory-Manager-Prozesse (siehe Abbildung 5.14). Jeder Memory-Manager speichert während der gesamten Rechnung statisch die Gittergeometrie- und Fluiddaten einer bestimmten Partition. Der Memory-Manager führt jedoch selber keine Trajektorienberechnungen durch. Die eigentliche Berechnung wird wie im DD-Verfahren von den Node-Prozessen ausgeführt. Im Unterschied zum DD-Verfahren kann ein Node-Prozeß Partikeltrajektorien prinzipiell in jeder beliebigen Partition berechnen. Die dazu jeweils aktuell benötigten Gitter- und Fluiddaten werden von dem Memory-Manager bereitgestellt, der diese Daten gespeichert hat. Wie im DD-Verfahren hat der Host-Prozeß auch hier die Aufgabe, die Partikelstartbedingungen zu verwalten und an die Nodes zu verteilen.

Im folgenden seien die Arbeitsweise und das Zusammenwirken der einzelnen Prozesse näher beschrieben:

Host:

Der Host verwaltet und verteilt die Partikelstartbedingungen. Dabei ist ei-

ne Startbedingung, wie schon oben erläutert, eine Menge von Daten, die den Partikelzustand beim Eintritt in das Strömungsgebiet bzw. beim Eintritt in eine neue Partition beschreiben. Der Host sendet eine Startbedingung an einen Node, und nach der Berechnung des entsprechenden Trajektoriensegmentes empfängt er von diesem Node den Endzustand der Trajektorie. Falls die Trajektorie eine Partitions-grenze erreicht hat, wird der Endzustand als neue Startbedingung behandelt und erneut an einen Node zur weiteren Berechnung gesendet. Dieser Vorgang wird vom Host solange wiederholt, bis alle Trajektorien endgültige Abbruchbedingungen erfüllt haben. Die Zuteilung von Startbedingungen an die Node-Prozesse unterliegt einem Optimierungsverfahren. Wie oben bereits erwähnt, kann ein Node prinzipiell in jeder Partition Trajektorien berechnen, indem er die entsprechenden Gitter- und Fluid-daten dieser Partition vom zugehörigen Memory-Manager empfängt. Um jedoch die Menge der zwischen den Prozessen auszutauschenden Daten und damit die Kommunikationszeit zu minimieren, sendet der Host eine Startbedingung bevorzugt an einen Node, in dessen Speicher bereits die Daten für die Partition vorliegen, in der diese Startbedingung lokalisiert ist.

Im Fall einer kollisionsbehafteten Simulation unter Verwendung des ITB-Verfahrens fallen diese Aufgaben innerhalb jeder Iteration an. Hierbei ist zu beachten, daß zu Beginn jeder neuen Iteration wiederum die ursprünglichen Startbedingungen, d.h. die Partikelzustände beim Eintritt in das Strömungsgebiet, benötigt werden. Diese muß der Host also permanent speichern. Im kollisionsfreien Fall wird dagegen jede initiale Startbedingung nur einmal am Anfang der Berechnung benötigt, d.h. sie kann aus dem Speicher des Hosts gelöscht werden, nachdem sie an einen Node-Prozeß zur Berechnung des zugehörigen Trajektoriensegmentes gesendet worden ist. Dies bedeutet, daß bei der Berechnung einer kollisionsbehafteten Strömung für den Host-Prozeß ein größerer Speicheraufwand anfällt. Dieser ist jedoch im allgemeinen nicht kritisch, weil der Speicherplatzbedarf für einige zehntausend Startbedingungen nur wenige MegaByte beträgt, und der Host neben den Startbedingungen keine größeren Datenmengen speichern muß.

Eine zusätzliche Aufgabe, die der Host in einer kollisionsbehafteten Simulation ausführen muß, ist die Überprüfung der Konvergenz des iterativen Algorithmus. Dazu müssen zunächst die Memory-Manager die Konvergenz lokal in den ihnen zugeordneten Partitionen überprüfen und diese Information an den Host senden. Der Host empfängt diese Informationen und entscheidet, ob global ein konvergenter Strömungszustand vorliegt.

Node:

Der Node empfängt jeweils eine Partikelstartbedingung vom Host, berechnet das zugehörige Trajektoriensegment und sendet den Endzustand des Segmen-

tes zurück an den Host. Nach dem Empfang der Startbedingung wird geprüft, ob die zugehörigen Startkoordinaten in der Gitterpartition liegen, deren Gitter- und Fluiddaten aktuell im lokalen Speicher vorhanden sind. Ist dies der Fall, wird sofort mit der Trajektorienberechnung begonnen. Im anderen Fall müssen diese Daten vom entsprechenden Memory-Manager zum Node übertragen werden. Zur Verfolgung eines Partikels im Strömungsgebiet und zur Detektierung der Annäherung an Partitions Grenzen müssen dem Node die kompletten Gittergeometriedaten der aktuellen Partition zur Verfügung stehen. Dies gilt jedoch nicht für die Fluiddaten. Es werden immer nur jeweils die Strömungsgrößen für die Gitterzelle benötigt, in der sich das Partikel gerade aufhält. Bezüglich der Übertragung der Fluiddaten vom Memory-Manager zum Node ergeben sich daraus zwei mögliche Strategien:

1. Übertragung der Daten für eine einzelne Gitterzelle, wenn das Partikel in diese Zelle eintritt. Sobald das Partikel in eine andere Zelle eintritt, werden die Strömungsgrößen für die nächste Zelle beim Memory-Manager angefordert. Bei Anwendung dieser Verfahrensweise ist die Menge der vom Memory-Manager an den Node zu übertragenden Daten minimal, jedoch ist die Anzahl der Kommunikationsvorgänge sehr groß.
2. Übertragung der Fluiddaten für die komplette Partition zu Beginn der Berechnung des Trajektoriensegmentes. Hierbei ist die Menge der zu übertragenden Daten maximal, jedoch ist nur ein einmaliger Kommunikationsvorgang notwendig.

Zwischen diesen beiden Strategien sind Abstufungen möglich, wie z.B. die Übertragung der Fluiddaten nicht für eine einzelne Gitterzelle, sondern für eine bestimmte Umgebung dieser Zelle. Eine allgemeingültige Beurteilung der Eignung dieser verschiedenen Strategien ist nicht möglich. Diese ist in hohem Maße von der Kommunikationsgeschwindigkeit und der Größe des lokalen Hauptspeichers der jeweils eingesetzten Rechner-Hardware abhängig [18].

Zur Berechnung von Partikel-Partikel-Kollisionen müssen die Partikelvariablen $u_{P0}, v_{P0}, \omega_{Px0}, \omega_{Py0}, d_{P0}$ und n_P aus der vorhergehenden Iteration des ITB-Algorithmus vorliegen. Wie auch die Fluidgrößen werden diese Partikelgrößen jeweils für die Gitterzelle benötigt, in der sich das Partikel aktuell aufhält. Die Partikeldaten der vorhergehenden Iteration werden von den Memory-Manager-Prozessen für die ihnen zugeordnete Partition gespeichert. Für die Übertragung der Partikeldaten von den Memory-Manager-Prozessen zu den Nodes gelten dieselben Anmerkungen, die oben für die Übertragung der Fluiddaten gemacht wurden.

Während der Berechnung eines Trajektoriensegmentes berechnet der Node für alle Gitterzellen, die durchquert werden, den Beitrag dieses Segmentes zu den

lokalen Feldgrößen der Partikelphase für die aktuelle Iteration. Anschließend werden diese neu ermittelten Partikeldaten an den zugehörigen Memory-Manager gesendet.

Memory-Manager:

Ein Memory-Manager speichert die Gittergeometrie- und Fluiddaten einer bestimmten Partition permanent während der gesamten Rechnung. Wenn ein Node die Gitter- bzw. Fluiddaten für diese Partition anfordert, werden sie an den entsprechenden Node gesendet. Der Memory-Manager führt selber keine Berechnungen aus. Seine Hauptaufgabe ist das Versenden von angeforderten Daten.

In einer kollisionsbehafteten Simulation entsteht für den Memory-Manager ein zusätzlicher Speicher- sowie Rechenaufwand. Für die ihm zugeordnete Partition müssen für jede Gitterzelle die Partikelvariablen u_{P0} , v_{P0} , ω_{Px0} , ω_{Py0} , d_{P0} und n_P gespeichert werden, und zwar sowohl die der vorhergehenden als auch der aktuellen Iteration. Im Falle einer Datenanforderung durch einen Node versendet der Memory-Manager zusätzlich zu den Gitter- und Fluiddaten die entsprechenden Partikelgrößen der vorhergehenden Iteration. Nach Beendigung der Trajektorienberechnung durch den Node empfängt der Memory-Manager von diesem die neu berechneten Partikeldaten und addiert diese Anteile zu den Gesamtpartikeldaten der aktuellen Iteration.

Durch die dynamische Zuordnung von Partitionen zu Node-Prozessen wird mit dem DSM-Verfahren eine sehr gute Verteilung der Rechenlast unter den Node-Prozessen erreicht [18, 55]. Ist der Rechenaufwand in einer Partition sehr groß, so bietet das DSM-Verfahren die Möglichkeit, daß mehrere Node-Prozesse gleichzeitig Trajektoriensegmente in dieser Partition berechnen. Für das in der Abbildung 5.15 dargestellte Beispiel der Strömung entlang einer rückwärts springenden Stufe bedeutet dies, daß der große Rechenaufwand, der in der mittleren Partition aufgrund des Rezirkulationsgebietes anfällt, auf die Nodes 2 und 3 verteilt wird. Hierzu muß der Node 3 die Gitter-, Fluid- sowie gegebenenfalls die Partikeldaten der mittleren Partition vom zugehörigen Memory-Manager 2 anfordern.

Im Vergleich zum DD-Verfahren ist der Kommunikationsaufwand des DSM-Verfahrens wesentlich größer, da hier nicht nur Partikelstartbedingungen, sondern außerdem auch Gittergeometrie- und Fluiddaten ausgetauscht werden müssen. Für kollisionsbehaftete Simulationen erhöht sich der Kommunikationsaufwand zusätzlich durch den notwendigen Austausch der Feldgrößen der Partikelphase.

Ein Nachteil des DSM-Verfahrens besteht darin, daß es nur auf solchen Rechnern effizient eingesetzt werden kann, die die gleichzeitige Aktivität mehrerer Prozesse

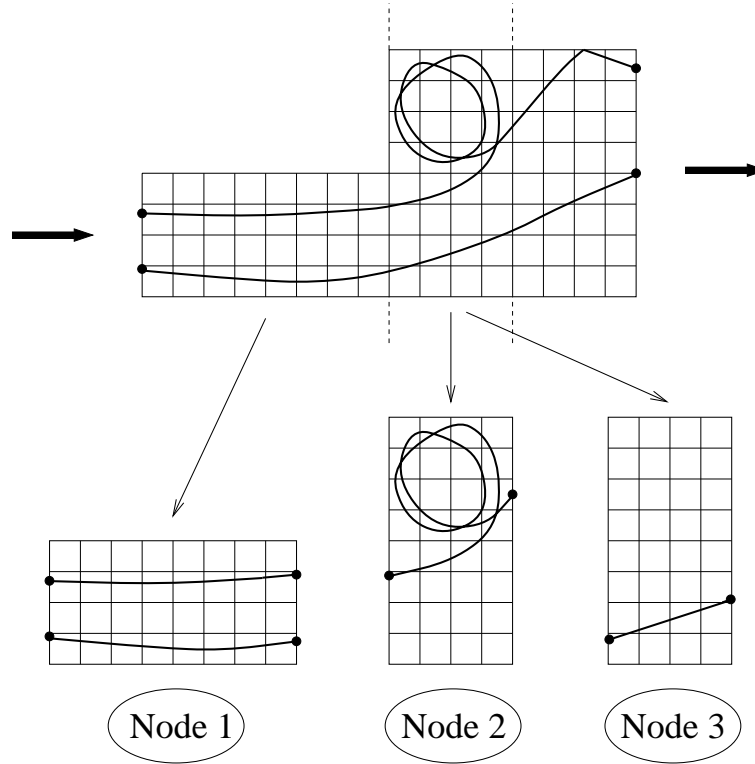


Abbildung 5.15: Beispiel für die Trajektorienberechnung im Distributed-Shared-Memory-Verfahren

auf einem Prozessor des Rechners zulassen. Durch die zusätzliche Verwendung der Memory-Manager-Prozesse werden wesentlich mehr Prozesse benötigt als Gitterpartitionen vorhanden sind. Im allgemeinen ist die Anzahl der benötigten Prozesse gleich $2N + 1$, wobei N die Anzahl der Partitionen ist. Falls alle Prozesse auf separaten Prozessoren laufen, müssen für die Simulation der Gas-Partikel-Strömung $2N + 1$ Prozessoren des Parallelrechners verwendet werden. Dies ist sehr ungünstig, da die eigentliche Hauptaufgabe der Simulation, nämlich die Berechnung der Partikeltrajektorien, nur von den N Node-Prozessen ausgeführt wird. Die übrigen $N + 1$ Prozesse, Memory-Manager und Host, dienen lediglich zur optimalen Verteilung der Rechenlast unter den Nodes. Ist es dagegen möglich, einen Prozessor des Parallelrechners mit jeweils einem Node- und einem Memory-Manager-Prozeß zu belegen, reduziert sich die Anzahl der benötigten Prozessoren um N . Die gegenseitige Beeinflussung der beiden Prozesse ist dabei gering, da sie einen komplementären Ressourcenbedarf haben. Der Node-Prozeß benötigt relativ wenig Speicherplatz und viel Rechenleistung, wogegen der Memory-Manager-Prozeß relativ viel Speicherplatz und nahezu

keine Rechenleistung beansprucht.

5.4.3 Neuer paralleler Algorithmus für die Simulation auf der Basis der simultanen Partikelverfolgung

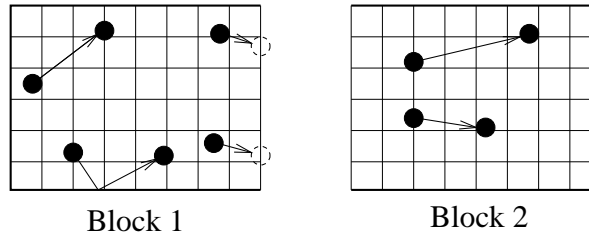
Das im Rahmen dieser Arbeit entwickelte Verfahren zur parallelen Berechnung der simultanen Partikelbewegung basiert auf der Methode der Gebietszerlegung. Das Strömungsgebiet wird in Partitionen aufgeteilt, wobei dieselbe Aufteilung verwendet wird, die zuvor zur Berechnung der Gasströmung verwendet wurde. Jede Partition wird einem Prozessor des Parallelrechners zugeordnet. Diese Zuordnung ist statisch, d.h. sämtliche Gittergeometrie- und Fluiddaten sind permanent während der gesamten Simulation in diesem Prozessor gespeichert. Die Bewegung aller simulierten Partikel, die sich zu einem bestimmten Zeitpunkt in einem Teilgebiet aufhalten, wird von dem Prozessor berechnet, dem dieses Teilgebiet zugeordnet ist. Im Gegensatz zu den parallelen Verfahren für die trajektorienbasierte Simulation existiert hier lediglich eine Sorte von Prozessen, nämlich Node-Prozesse. Jeder Node-Prozeß arbeitet auf einem separaten Prozessor. Die Anzahl der Prozesse bzw. Prozessoren ist gleich der Anzahl der Gitterpartitionen. Jeder Node kann mit jedem anderen Node Informationen austauschen, und alle Nodes führen die gleichen Aufgaben aus.

Das Prinzip des parallelen Verfahrens für die simultane Partikelverfolgung ist exemplarisch in der Abbildung 5.16 bzw. als Ablaufdiagramm in der Abbildung 5.17 dargestellt. Vor der Ausführung eines Zeitschrittes wird zunächst überprüft, ob die Simulation aufgrund der Erfüllung eines Abbruchkriteriums beendet werden kann. Ein mögliches Abbruchkriterium für die Simulation einer instationären Strömung ist z.B. das Erreichen einer vorgegebenen physikalischen Zeitschranke. Bei der Simulation einer stationären Strömung kann das Erreichen eines stationären Strömungszustands als Abbruchbedingung dienen.

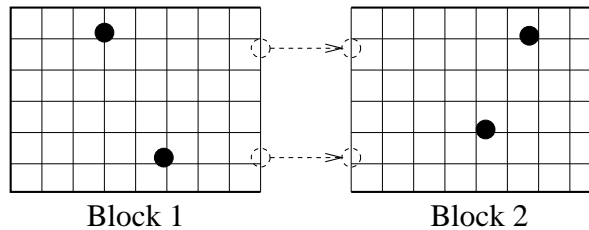
Falls kein Abbruchkriterium erfüllt ist, erfolgt die Ausführung des nächsten Zeitschrittes Δt . Dazu wird zunächst die kollisionsfreie Bewegung aller Partikel während dieses Zeitschrittes berechnet. Dabei bestimmt jeder Node nur die Bewegung der Partikel, die sich in der ihm zugeordneten Partition befinden. Falls ein Partikel innerhalb von Δt gegen eine Strömungsberandung stößt, wird der Partikel-Wand-Stoß nach dem im Abschnitt 3.3.1 beschriebenen Verfahren ausgeführt, und die Berechnung der Partikelbewegung wird für den verbleibenden Teil von Δt fortgesetzt.

Trifft ein Partikel während des Zeitschrittes auf eine Partitions-grenze, so kann die Berechnung für dieses Partikel zunächst nicht fortgesetzt werden. Hierzu sind Informationen über die Gittergeometrie und das Gasgeschwindigkeitsfeld jenseits der Partitions-grenze erforderlich, die für den aktuellen Node jedoch nicht verfügbar sind.

1. Ausführen des Zeitschrittes Δt



2. Übertragen der Partikel an den Blockrändern



3. Ausführen des verbleibenden Teilschrittes

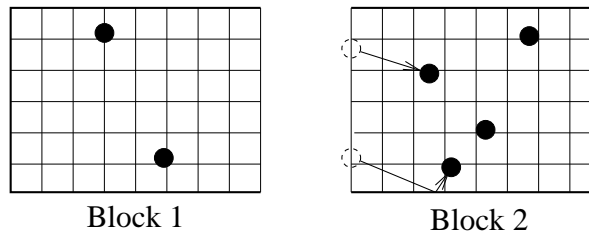


Abbildung 5.16: Schema der parallelen simultanen Partikelverfolgung für einen Zeitschritt Δt

Die weitere Berechnung der Bewegung dieses Partikels muß also von dem Node durchgeführt werden, dem das benachbarte Teilgebiet zugeordnet ist. Zu diesem Zweck müssen die Daten dieses Partikels an den betreffenden Node übertragen werden. Um die Gesamtzahl der Kommunikationsvorgänge zu minimieren, werden die Partikeldaten nicht sofort übertragen, sondern zunächst in einem Buffer zwischengespeichert. In diesem Buffer werden die Daten aller Partikel gespeichert, die während des Zeitschrittes Δt dieselbe Partitions-grenze erreichen. Erst nachdem für alle Partikel innerhalb des Teilgebietes der aktuelle Schritt ausgeführt wurde, wird der komplette Buffer an den für das Nachbar-teilgebiet zuständigen Node gesendet. Dieser führt die Berechnung der Partikelbewegung für den jeweils verbleibenden Teil des Zeitschrittes zu Ende. Hierbei ist zu beachten, daß das Partikel auch im Verlauf dieses Restschrittes wiederum eine Partitions-grenze erreichen kann. In diesem Fall

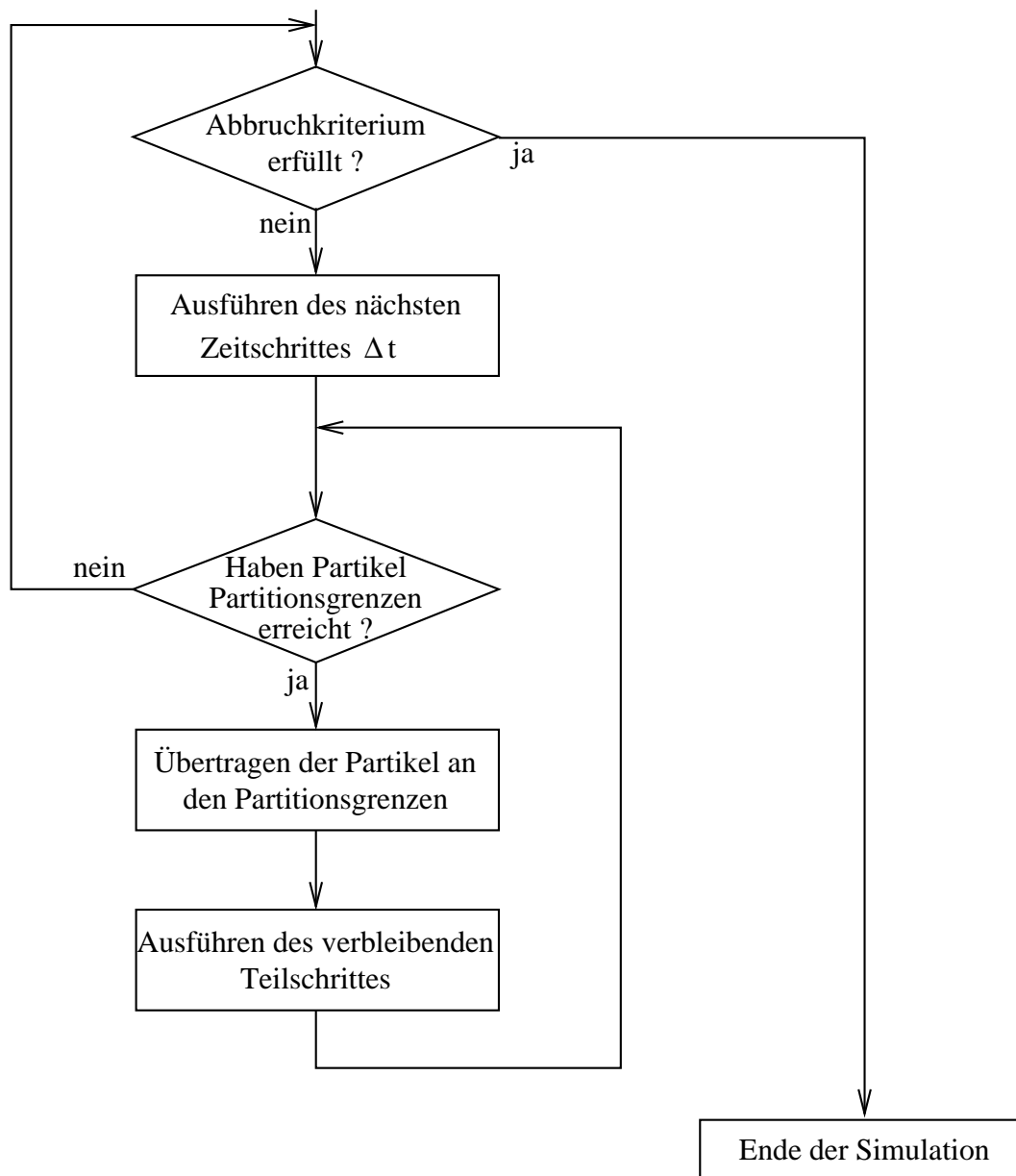


Abbildung 5.17: Ablaufdiagramm für einen Zeitschritt der parallelen simultanen Partikelverfolgung

ist eine weitere Übertragung der Partikeldaten notwendig.

Im Fall der Berücksichtigung von Partikel–Partikel–Kollisionen werden diese Kollisionen im Anschluß an die kollisionsfreie Partikelbewegung im Zeitschritt Δt berechnet. Hierzu wird für jedes Partikel mit Hilfe des im Abschnitt 4.4.2 beschriebenen Monte–Carlo–Verfahrens überprüft, ob eine Kollision stattfindet und diese gegebenenfalls ausgeführt. Des weiteren wird im kollisionsbehafteten Fall nach jedem Zeitschritt die maximale lokale Kollisionsfrequenz $\nu_{C,max}$ für das gesamte Strömungsgebiet bestimmt. Der Kehrwert von $\nu_{C,max}$ ist die minimale Kollisionszeit $\tau_{C,min}$. Diese bestimmt die Größe des nachfolgenden Zeitschrittes Δt , für den $\Delta t \ll \tau_{C,min}$ gelten muß.

Ein Vorteil des hier verwendeten parallelen Verfahrens für die simultane Partikelverfolgung besteht darin, daß sämtliche zu einer Gitterpartition gehörenden Daten permanent in einem Prozeß gespeichert sind. Dieser Prozeß berechnet zunächst die Gasströmung in dem ihm zugeordneten Teilgebiet. Anschließend erfolgt in demselben Teilgebiet die Berechnung der Partikelbewegung. Die dazu benötigten Gittergeometrie– und Fluidaten sind zu Beginn der Partikelberechnung also bereits lokal vorhanden, so daß kein Kommunikationsaufwand zur Beschaffung dieser Daten anfällt. Der Datenaustausch zwischen den Prozessen ist beschränkt auf die Übertragung der Daten der Partikel, die während eines Zeitschrittes eine Partitionsgränze überqueren. Die Gesamtmenge der auszutauschenden Daten ist daher in diesem Verfahren relativ klein.

Ähnlich wie das Domain–Decomposition–Verfahren für die trajektorienbasierte Simulation hat das in diesem Abschnitt beschriebene Verfahren den Nachteil, daß die Rechenlastverteilung unter den Prozessen sehr ungünstig sein kann. Der Rechenaufwand, der in einem Teilgebiet zu einem bestimmten Zeitpunkt anfällt, ist im wesentlichen von der Anzahl der Partikel abhängig, die sich zu diesem Zeitpunkt in dem Teilgebiet aufhalten. Eine gleichmäßige Verteilung der Rechenlast ist also nur dann zu erzielen, wenn sich in allen Partitionen ungefähr gleich viele Partikel aufhalten. Dies ist jedoch im allgemeinen nicht der Fall. Des weiteren ist zu beachten, daß die Simulation mit Hilfe der simultanen Partikelverfolgung inhärent instationär ist. Dies bedeutet, daß sich in der Regel die Partikelverteilung im Strömungsgebiet im Verlauf der Simulation verändert. Im Gegensatz dazu ist jedoch die Gitterpartitionierung statisch, so daß sich eine optimale Rechenlastverteilung zu jedem Zeitpunkt nicht erreichen läßt. Eine dynamische, an die jeweilige Partikelverteilung angepaßte Repartitionierung des Gitternetzes wäre für die hier verwendeten blockstrukturierten Gitter algorithmisch äußerst aufwendig und mit einem hohen Kommunikationsaufwand verbunden. Eine mögliche Verbesserung dieses Verfahrens bietet die Verwendung unstrukturierter Gitternetze, mit Hilfe derer eine Repartitionierung in Abhängigkeit von der Rechenlast wesentlich einfacher ist. Dies kann Gegenstand zukünftiger Untersuchungen sein.

Kapitel 6

Ergebnisse der Testrechnungen

Zur Untersuchung der im Kapitel 5 beschriebenen parallelen Simulationsalgorithmen wurden Gas-Partikel-Strömungen in zwei unterschiedlichen Testgeometrien betrachtet, und zwar in einem vertikalen und in einem horizontalen Kanal. Die Strömungen wurden dabei als zweidimensional angesehen. Dies hat den Nachteil, daß die Simulationen die physikalische Realität nicht exakt wiedergeben können, da insbesondere kollisionsbehaftete Strömungen immer einen stark dreidimensionalen Charakter haben. Diesem Nachteil steht jedoch eine Reihe von Vorteilen gegenüber. Für die Untersuchung der parallelen Algorithmen ist eine Vielzahl von Testrechnungen mit unterschiedlichen Parametern erforderlich. Durch die Beschränkung auf zweidimensionale Geometrien kann der hierfür notwendige Rechenzeitaufwand erheblich verringert werden. Außerdem sind die Zusammenhänge zwischen den physikalischen Vorgängen in einer Strömung und der Leistungsfähigkeit der Simulationsalgorithmen für eine zweidimensionale Strömung wesentlich einfacher zu erfassen. Des weiteren können auch mit Hilfe einer zweidimensionalen Simulation die wichtigsten physikalischen Effekte, die eine Strömung dominieren, zumindest qualitativ erfaßt werden. Aus diesen Gründen wurde für den Zweck der grundlegenden Untersuchung der parallelen Algorithmen die Betrachtung zweidimensionaler Strömungen als ausreichend angesehen.

In den folgenden Abschnitten 6.1 und 6.2 werden die Ergebnisse der Testrechnungen für die beiden Geometrien vorgestellt. Dabei erfolgt zunächst jeweils die Beschreibung der physikalischen Ergebnisse der Strömungsberechnungen. Diese werden verglichen mit Ergebnissen, die der Literatur entnommen wurden. Im Anschluß daran werden jeweils die Untersuchungen zur Leistungsfähigkeit der verschiedenen parallelen Simulationsalgorithmen vorgestellt.

Die Rechnungen wurden auf dem Parallelrechner Cray T3D des Edinburgh Parallel Computing Centre (EPCC) durchgeführt. Dieser Rechner besteht aus 512 Prozes-

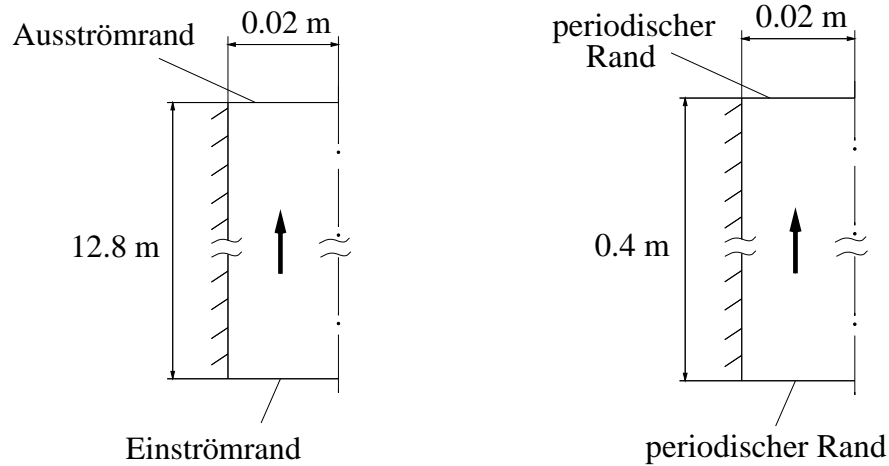


Abbildung 6.1: Strömungsgeometrie für die Simulation der vertikalen Kanalströmung für die trajektorienbasierte Simulation (links) bzw. für die simultane Partikelverfolgung (rechts)

soren, die über einen lokalen Speicher von jeweils 64 MB verfügen.

6.1 Berechnung der Strömung in einem vertikalen Kanal

6.1.1 Physikalische Eigenschaften der Strömung

Als erstes Testbeispiel wurde eine aufwärts gerichtete Gas-Partikel-Strömung in einem vertikalen Kanal untersucht. Die Strömung kann als symmetrisch bezüglich der Kanalachse angesehen werden. Aus diesem Grund wurde die Strömungsberechnung jeweils nur für eine Hälfte des Kanals durchgeführt und in der Mitte des Kanals eine Symmetrierandbedingung verwendet. Die Abmaße des Strömungsgebietes sind in der Abbildung 6.1 dargestellt. Für die trajektorienbasierte Simulation hat der Kanal am unteren Ende einen Einströmrand, am oberen Ende einen Ausströmrand und eine Gesamtlänge von 12.8 m. Mit Hilfe von vorab durchgeführten Testrechnungen wurde ermittelt, daß diese Lauflänge erforderlich ist, damit am oberen Ende des Kanals für die Partikelphase ein ausgebildeter Strömungszustand vorliegt. Für die Simulation mit Hilfe der simultanen Partikelverfolgung wurde eine 0.4 m lange Kanalgeometrie mit periodischen Randbedingungen am oberen und unteren Rand verwendet. Dies bedeutet, daß jedes Partikel, das die Geometrie durch einen dieser

Ränder verläßt, gleichzeitig am jeweils gegenüberliegenden Rand wieder in die Geometrie eintritt. Für beide Simulationsmethoden bestand die linke Begrenzung des Strömungsgebietes aus einer festen Wand und die rechte aus einem Symmetrierand, der die Kanalachse repräsentiert.

Von Tanaka und Tsuji [46] bzw. Tanaka et al. [45] wurden Simulationen bzw. Messungen für eine vertikale Rohrströmung zur Untersuchung des Einflusses der Partikel–Partikel–Kollisionen durchgeführt. Obwohl es sich dabei um eine dreidimensionale Strömung handelt, sind die Ergebnisse dieser Untersuchungen dazu geeignet, sie mit den im Rahmen der vorliegenden Arbeit erzielten Simulationsergebnissen für die vertikale Kanalströmung zumindest qualitativ zu vergleichen. Die in dieser Arbeit verwendeten physikalischen Parameter der Gas–Partikel–Strömung wurden deshalb weitgehend von Tanaka und Tsuji [46] übernommen. Als Gasphase wurde Luft verwendet, in der Partikel aus Polystyren transportiert wurden. Die Kanalwand bestand, analog zu den Experimenten von Tanaka et al. [45], aus Acryl. Die hier verwendeten Werte der wichtigsten physikalischen Parameter sind in der Tabelle 6.1 zusammengefaßt.

Die Berechnung der Gasströmung und die Berechnung der Partikelbewegung wurden getrennt voneinander durchgeführt. Es wurde angenommen, daß die Strömungsgrößen des Gases, d.h. die Geschwindigkeitskomponenten und die Turbulenzgrößen, für beide Kanalgeometrien über die gesamte Länge des Kanals konstant sind und denen einer ausgebildeten Gasströmung entsprechen. Die Rückwirkung der Partikel auf die Gasströmung wurde hier nicht berücksichtigt. Dazu ist anzumerken, daß gerade die in dieser Arbeit betrachteten Strömungen mit mittlerer bis hoher Partikelbeladung, in denen Inter–Partikel–Kollisionen eine wichtige Rolle spielen, in der Praxis auch immer eine starke Phasenwechselwirkung aufweisen. Eine Vernachlässigung dieses Effektes bedeutet also ein Abweichen von der physikalischen Realität. Andererseits ist, wie oben bereits erwähnt, das Hauptziel dieser Arbeit die Untersuchung paralleler Simulationsalgorithmen. Die Berücksichtigung der Rückwirkung der Partikelbewegung auf das Gas brächte in dieser Hinsicht keinen zusätzlichen Erkenntnisgewinn, sondern erhöhte lediglich den erforderlichen Rechenaufwand. Des weiteren wurden die Simulationen von Tanaka und Tsuji [46] ebenfalls ohne die Einbeziehung der Phasenwechselwirkung ausgeführt, so daß deren Vernachlässigung eine bessere Vergleichbarkeit der Ergebnisse gewährleistet.

Es wurde jeweils eine Strömung mit einer niedrigen Partikelbeladung ($\alpha_P = 10^{-4}$, $\eta = 0.086$) und einer höheren Beladung ($\alpha_P = 5 \cdot 10^{-3}$, $\eta = 4.3$) berechnet. Erstere kann als verdünnte Strömung angenommen werden, weshalb Partikel–Partikel–Kollisionen in diesem Fall nicht berücksichtigt wurden. Dagegen wurde die Berechnung der höher beladenen Strömung unter Einbeziehung der Kollisionen durchgeführt. Jeder dieser beiden Testfälle wurde auf zwei verschiedene Arten berechnet, nämlich zum einen mit der Methode der Trajektorienberechnung und zum anderen

Physikalische Größe	Formel- zeichen	Wert
Kanalbreite	B	40 mm
Mittlere axiale Gasgeschwindigkeit	\bar{v}_F	16 m/s
Dichte des Gases	ρ_F	1.21 kg/m ³
Dynamische Viskosität des Gases	μ	$1.79 \cdot 10^{-5}$ kg/(m s)
Mittlerer Partikeldurchmesser	\bar{d}_P	406 μ m
Materialdichte der Partikel	ρ_P	1038 kg/m ³
Stoßzahl für Part.–Wand–Stoß	e_w	0.94
Reibungsbeiwert für Part.–Wand–Stoß	f_w	0.325
Stoßzahl für Part.–Part.–Stoß	e_p	0.94
Reibungsbeiwert für Part.–Part.–Stoß	f_p	0.325
Volumetrische Beladung	α_P	<div>ohne Koll.: 10^{-4}</div> <hr/> <div>mit Koll.: $5 \cdot 10^{-3}$</div>
Massenbeladung	η	<div>ohne Koll.: 0.086</div> <hr/> <div>mit Koll.: 4.3</div>

Tabelle 6.1: Physikalische Parameter für die Simulation der vertikalen Kanalströmung

mit Hilfe der simultanen Partikelverfolgung. Die dabei verwendeten Simulationsparameter sind im folgenden zusammengefaßt:

Trajektorienberechnung

1. Kollisionsfreie Strömung: Es wurden 5 000 Trajektorien vom Einströmrand bis zum Ausströmrand berechnet. Des weiteren wurden Testrechnungen mit größeren Anzahlen von Trajektorien durchgeführt, wobei sich jedoch keine Veränderungen in den Ergebnissen zeigten. Aufgrund der relativ einfachen Strömungsgeometrie ist die Zahl von 5 000 Trajektorien ausreichend, um die gemittelten Strömungsgrößen der Partikelphase auf statistisch gesicherte Weise zu bestimmen.
2. Kollisionsbehaftete Strömung: Wie im kollisionsfreien Fall wurden auch hier zunächst 5 000 Trajektorien verwendet. Zur Berücksichtigung der Kollisionen mit Hilfe der Iterierten Trajektorienberechnung wurden zehn aufeinanderfolgende Trajektorienberechnungen durchgeführt. Wie bereits im Abschnitt 4.4.1 erwähnt, sind für dieses Verfahren bisher keine allgemeingültigen Angaben über die Anzahl der notwendigen Iterationen sowie über Konvergenzkriterien bekannt. Daher wurden zusätzliche Testrechnungen mit der doppelten Anzahl von Trajektorien sowie der doppelten Anzahl von Iterationen durchgeführt. Die dabei berechneten mittleren Strömungsgrößen für die Partikelphase waren identisch mit den zuvor berechneten. Die Kombination von 5 000 Trajektorien und 10 Iterationen kann daher für den vorliegenden Testfall als ausreichend angesehen werden.

Simultane Partikelverfolgung

1. Kollisionsfreie Strömung: Hierbei wurde die Bewegung von 10 000 Partikeln berechnet. Die Verwendung eines relativ kurzen Kanalsegmentes mit periodischen Randbedingungen am oberen und unteren Rand hat dabei den Vorteil, daß die Anzahl der simulierten Partikel während der gesamten Rechnung konstant bleibt. Dadurch läßt sich der Rechenzeitbedarf der Simulation leichter im voraus abschätzen. Es wäre auch denkbar, dieselbe Geometrie wie für die Trajektorienberechnung zu verwenden. In diesem Fall müßten in bestimmten Zeitabständen neue Partikel am Einströmrand eingesetzt werden, um einen konstanten Partikelmassenstrom zu gewährleisten. Dies bedeutete jedoch, daß sich die Gesamtzahl der Partikel im Strömungsgebiet laufend veränderte, was eine Rechenzeitabschätzung praktisch unmöglich machte.

Aus der oben beschriebenen Trajektorienberechnung ergab sich, daß die durchschnittliche Aufenthaltsdauer der Partikel im Kanal 0.6 s betrug. Entsprechend wurde diese Zeit als physikalische Laufzeit für die simultane Partikelverfolgung

verwendet. Die Größe der dabei ausgeführten Zeitschritte war in Abhängigkeit von den momentanen Genauigkeitsanforderungen veränderlich, jedoch wurde der minimal zugelassene Zeitschritt auf $\Delta t_{min} = 10^{-5}$ s begrenzt.

Die simultane Partikelverfolgung ist ein inhärent instationäres Verfahren, so daß am Ende einer solchen Simulation die Verteilung der Partikelgrößen zu genau diesem bestimmten Zeitpunkt vorliegt. Dagegen werden mit Hilfe der Trajektorienberechnung zeitlich gemittelte Werte bestimmt. Um die Ergebnisse der beiden Simulationsverfahren miteinander vergleichen zu können, wurde für die simultane Partikelverfolgung ebenfalls eine Mittelung durchgeführt. Hierzu wurden die Strömungsgrößen der Partikelphase jeweils für einen konstanten Wandabstand über die gesamte Länge des Kanalsegmentes gemittelt.

2. Kollisionsbehaftete Strömung: Auch für diese Simulation wurden 10 000 Partikel verwendet. Die physikalische Dauer der Simulation betrug 0.56 s, was etwas kürzer als die Dauer der kollisionsfreien Berechnung ist. Der Grund hierfür ist, daß im kollisionsbehafteten Fall die mittlere axiale Partikelgeschwindigkeit größer und damit die durchschnittliche Aufenthaltsdauer der Partikel im Kanal kleiner als im kollisionsfreien Fall ist. Die Partikel-Partikel-Kollisionen wurden mit Hilfe der im Abschnitt 4.4.2 beschriebenen Monte-Carlo-Methode berechnet. Für die Größe der Zeitschritte und die Berechnung der gemittelten Strömungsgrößen der Partikelphase gelten dieselben Angaben wie im kollisionsfreien Fall.

Die physikalischen Ergebnisse für die Simulation der vertikalen Kanalströmung sind in den Abbildungen 6.2 bis 6.5 dargestellt. In diesen Diagrammen bezeichnet die auf der horizontalen Achse aufgetragene Größe x jeweils den Abstand von der Kanalwand, d.h. der Wert $x = 0$ m bezeichnet einen Ort an der Wand, und der Wert $x = 0.02$ m bezeichnet die Mitte des Kanals. In der Abbildung 6.2 ist für den kollisionsfreien Fall der Verlauf der relativen Anzahldichte der Partikel n_P/n_{P0} über den Kanalquerschnitt dargestellt. Die relative Anzahldichte ist dabei das Verhältnis der lokalen zu der über den Querschnitt gemittelten Anzahldichte. Die Abbildung zeigt die Ergebnisse der im Rahmen der vorliegenden Arbeit durchgeführten Berechnungen sowie einer Simulation von Tanaka und Tsuji [46] für eine vertikale Rohrströmung. Man sieht, daß die Partikelkonzentration in der Mitte des Kanals relativ hoch ist und zur Wand hin rasch abnimmt. Dies ist auf die Wirkung der Saffman-Kraft zurückzuführen. Da die Partikel aufgrund der nach unten wirkenden Schwerkraft eine geringere axiale Geschwindigkeit als das umgebende Gas haben, ist der Gradient der Relativgeschwindigkeit zwischen Partikel und Gas und damit auch die Saffman-Kraft zur Kanalmitte hin gerichtet. Lediglich in unmittelbarer Wandnähe ist die Partikelgeschwindigkeit größer als die Gasgeschwindigkeit. Dies ist darauf zurückzuführen, daß ein Partikel, das sich aus dem Inneren des Strömungsge-

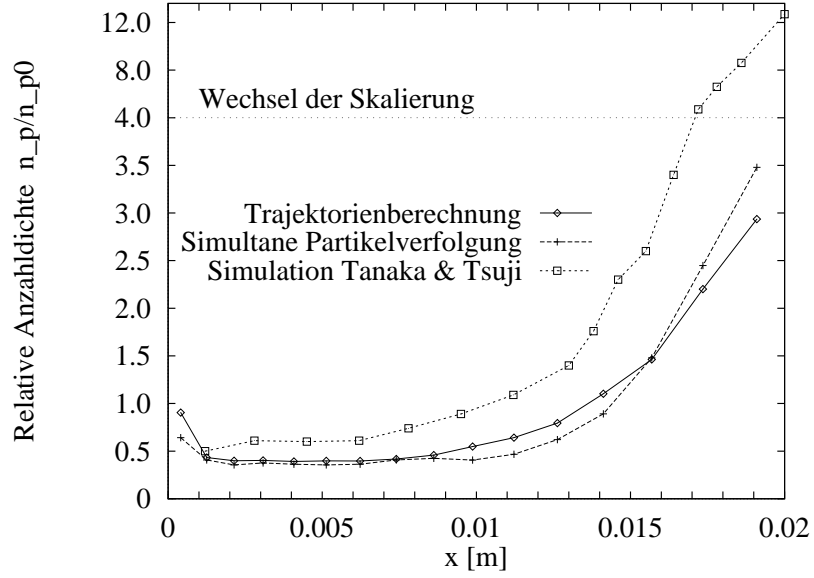


Abbildung 6.2: Relative Partikelanzahldichte n_P/n_{P0} für die kollisionsfreie Strömung ($\alpha_P = 10^{-4}$, $\eta = 0.086$)

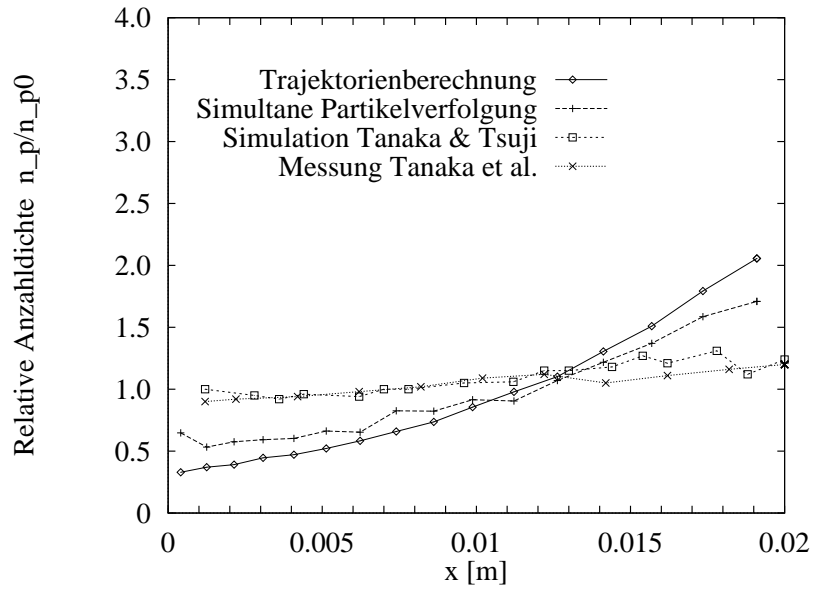


Abbildung 6.3: Relative Partikelanzahldichte n_P/n_{P0} für die kollisionsbehaftete Strömung ($\alpha_P = 5 \cdot 10^{-3}$, $\eta = 4.3$)

bietes der Wand annähert, sich nur für eine sehr kurze Zeit in der Wandgrenzschicht aufhält. Diese Zeit ist in der Regel wesentlich kürzer als die aerodynamische Relaxationszeit, so daß sich die Geschwindigkeit des Partikels nicht ändert. In der Nähe der Wand ist also die Saffman–Kraft zur Wand hin gerichtet. Aufgrund der hohen Rotationsgeschwindigkeiten, die die Partikel nach einem Wandstoß besitzen, spielt die Magnus–Kraft in diesem wandnahen Gebiet ebenfalls eine wichtige Rolle. Wie die Saffman–Kraft ist auch die Magnus–Kraft zur Wand hin gerichtet. Daher ist in unmittelbarer Wandnähe ein leichter Anstieg der Partikelkonzentration zu beobachten. Wie in der Abbildung 6.2 zu sehen ist, ergab sich aus der Simulation von Tanaka und Tsuji [46] in der Mitte des Strömungsgebietes eine wesentlich höhere Partikelkonzentration als bei den Simulationen dieser Arbeit. Dies ist dadurch begründet, daß Tanaka und Tsuji [46] in ihren Rechnungen den Einfluß der Fluidturbulenz auf die Partikelbewegung vernachlässigten. Dieser Effekt wurde jedoch in der vorliegenden Arbeit berücksichtigt, was eine stärkere Dispersion der Partikel zur Folge hat.

In der Abbildung 6.3 ist die Verteilung der relativen Anzahldichte für den kollisionsbehafteten Fall dargestellt. Diese Verteilung ist insgesamt wesentlich gleichmäßiger als im kollisionsfreien Fall. Die Partikel–Partikel–Kollisionen haben eine dispergierende Wirkung. Die mittlere Zeit zwischen zwei Kollisionen τ_C ist kleiner als die aerodynamische Relaxationszeit τ_A . Deshalb ist der Einfluß der Saffman–Kraft auf die Partikelbewegung geringer als in der kollisionsfreien Strömung. Die Simulation von Tanaka und Tsuji [46] und die Messung von Tanaka et al. [45] für die Rohrströmung zeigen einen nahezu konstanten Verlauf der Anzahldichte über den Rohrquerschnitt. Im Gegensatz dazu ergab sich aus den Simulationen dieser Arbeit für die Kanalströmung ein leichter Anstieg der Partikelkonzentration zur Kanalmitte hin. Durch die hier vorgenommene Beschränkung auf eine zweidimensionale Strömungsberechnung wird der dispergierende Charakter der Kollisionen abgeschwächt. Der grundsätzliche Effekt der verstärkten Dispersion der Partikel im Vergleich zum kollisionsfreien Fall ist jedoch auch für die zweidimensionale Berechnung zu beobachten.

Die Abbildungen 6.4 und 6.5 zeigen den Verlauf der axialen Partikelgeschwindigkeit für die kollisionsfreie bzw. die kollisionsbehaftete Strömung. Des weiteren ist in den Diagrammen zum Vergleich jeweils das Geschwindigkeitsprofil der Gasphase dargestellt. Für den kollisionsfreien Fall ist die maximale Partikelgeschwindigkeit in der Mitte des Kanals bzw. des Rohres zu beobachten, während sie zur Wand hin abnimmt. Dabei sinkt bei den im Rahmen dieser Arbeit durchgeführten Berechnungen die Geschwindigkeit sehr viel stärker ab als in der Simulation von Tanaka und Tsuji [46]. Dies ist wiederum darauf zurückzuführen, daß letztere den Einfluß der Fluidturbulenz auf die Partikelbewegung vernachlässigten. Die turbulenten Schwankungsbewegungen des Gases forcieren die horizontale Komponente der Partikelbewegung. Dadurch kollidiert ein Partikel häufiger mit den Wänden, wobei jedesmal die Partikelgeschwindigkeit reduziert wird. Nach einem solchen Wandstoß

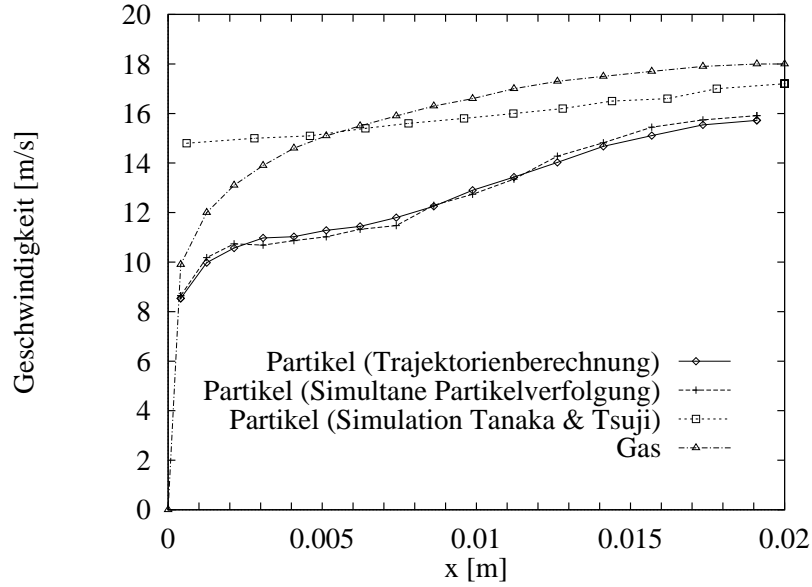


Abbildung 6.4: Mittlere axiale Geschwindigkeit für die Gas- und Partikelphase für die kollisionsfreie Strömung ($\alpha_P = 10^{-4}$, $\eta = 0.086$)

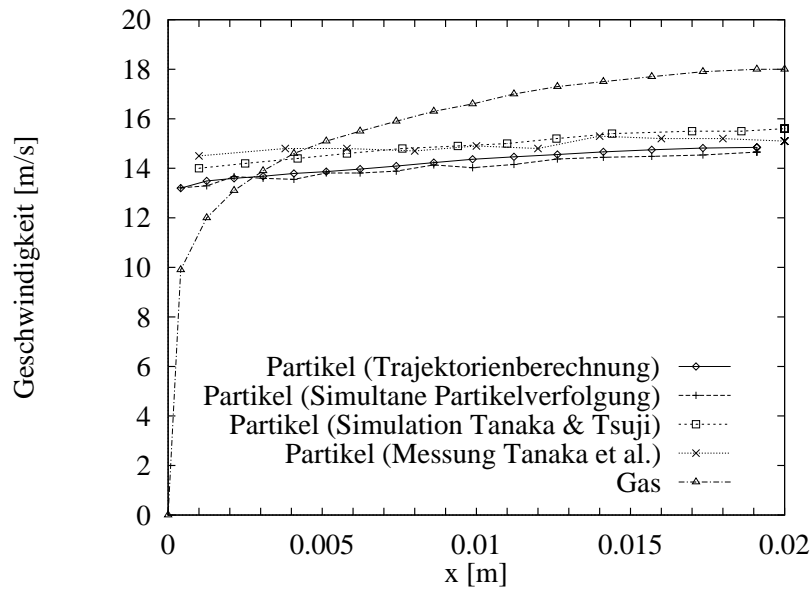


Abbildung 6.5: Mittlere axiale Geschwindigkeit für die Gas- und Partikelphase für die kollisionsbehaftete Strömung ($\alpha_P = 5 \cdot 10^{-3}$, $\eta = 4.3$)

bewegt sich das Partikel zurück in das Innere des Strömungsgebietes, wo es durch die Gasströmung wiederum beschleunigt wird. Die Berücksichtigung des Einflusses der Fluidturbulenz hat also zur Folge, daß vermehrt Wandstöße stattfinden und die mittlere Partikelgeschwindigkeit zur Wand hin deutlich abnimmt.

Im kollisionsbehafteten Fall, der in der Abbildung 6.5 dargestellt ist, ist die mittlere axiale Geschwindigkeit der Partikel nahezu konstant über den Querschnitt. Dies ist durch den intensiven Impulsaustausch zwischen den Partikeln begründet, der durch die Kollisionen verursacht wird. Im Gegensatz zum kollisionsfreien Fall zeigt sich eine gute Übereinstimmung zwischen den im Rahmen dieser Arbeit berechneten Geschwindigkeitsprofilen und den Ergebnissen der Simulation Tanakas und Tsujis [46] bzw. der Experimente Tanakas et al. [45]. Dies unterstreicht, daß die aerodynamischen Effekte, wie z.B. der Einfluß der Fluidturbulenz, in der kollisionsbehafteten Strömung nur eine untergeordnete Rolle spielen und der Charakter der Strömung vor allem durch die Partikel-Partikel-Kollisionen geprägt wird.

6.1.2 Anwendung der parallelen Simulationsalgorithmen auf die Berechnung der vertikalen Kanalströmung

Parallele Trajektorienberechnung

Zur Untersuchung der parallelen Verfahren für die Trajektorienberechnung, d.h. der DD- und der DSM-Methode, wurde zunächst ein serieller Testfall definiert. Die Ergebnisse der seriellen Berechnung dienen als Referenzgrößen zur Bestimmung des Speedups und der Effizienz für die parallele Berechnung. Das Gitternetz für die Kanalgeometrie bestand im seriellen Fall aus 1024 Gitterzellen in Richtung der Kanalachse und 16 Zellen in horizontaler Richtung. Es wurde jeweils eine kollisionsfreie und eine kollisionsbehaftete Trajektorienberechnung durchgeführt:

1. Kollisionsfreier Testfall: In diesem Fall wurden 650 Partikeltrajektorien berechnet. Die dafür benötigte Rechenzeit ist die Vergleichsgröße für die kollisionsfreien parallelen Testrechnungen.
2. Kollisionsbehafteter Testfall: Hierbei wurden 400 Trajektorien verwendet. Zur Berücksichtigung der Kollisionen wurden zwei Iterationen des ITB-Algorithmus ausgeführt. Die Rechenzeit für die zweite, kollisionsbehaftete Iteration bildete die Vergleichsgröße zur Beurteilung der parallelen Verfahren.

Diese beiden Testfälle wurden sowohl seriell als auch parallel berechnet. Für die parallele Berechnung wurde das Gitternetz in verschiedene Anzahlen von Partitionen

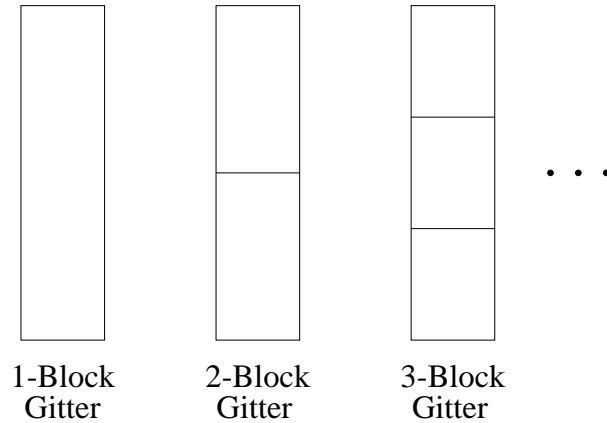


Abbildung 6.6: Aufteilung des Gitters für den vertikalen Kanal in Partitionen für die parallele Trajektorienberechnung

aufgeteilt, nämlich 2, 3, 7, 15, 31 sowie 63 Partitionen. Die Wahl dieser Partitionszahlen wurde zum einen bestimmt durch die Anforderungen der parallelen Verfahren (DD,DSM), zum anderen durch Regelungen bei der Nutzung des verwendeten Parallelrechners (Cray T3D am EPCC). Betrachtet man eine bestimmte Partitionsanzahl N , so werden innerhalb des DD-Verfahrens $N + 1$ und innerhalb des DSM-Verfahrens $2N + 1$ Prozesse für die parallele Berechnung benötigt. Andererseits kann auf dem hier eingesetzten Rechner die Anzahl von Prozessoren, die für einen Programmlauf verwendet wird, nicht beliebig gewählt werden, sondern muß eine Potenz der Zahl 2 sein. Des weiteren kann auf jedem Prozessor jeweils nur ein Prozess aktiv sein. Durch die Wahl der Partitionsanzahlen in der oben beschriebenen Weise wird daher eine optimale Ausnutzung der Rechnerressourcen gewährleistet.

Die Gitterpartitionierung wurde derart durchgeführt, daß die resultierenden Teilgebiete denselben Flächeninhalt sowie dieselbe Anzahl von Gitterzellen besaßen. Um außerdem die Anzahl der Gitterzellen an den Partitionsrändern zu minimieren, wurde die Kanalgeometrie ausschließlich in vertikal übereinander liegende Segmente unterteilt, wie in der Abbildung 6.6 gezeigt.

In den Abbildungen 6.7 und 6.8 ist der Verlauf des Speedups bzw. der Effizienz der parallelen Simulationsalgorithmen für die Trajektorienberechnung dargestellt. Bei der Bestimmung dieser Größen wurden ausschließlich die Ausführungszeiten der Node-Prozesse berücksichtigt, da diese die eigentliche Trajektorienberechnung durchführen. Der Vergleich zwischen dem Domain-Decomposition- und dem Distributed-Shared-Memory-Verfahren zeigt, daß sowohl im kollisionsfreien als auch im kollisionsbehafteten Fall mit letzterem Verfahren deutlich höhere Speedups bzw. Ef-

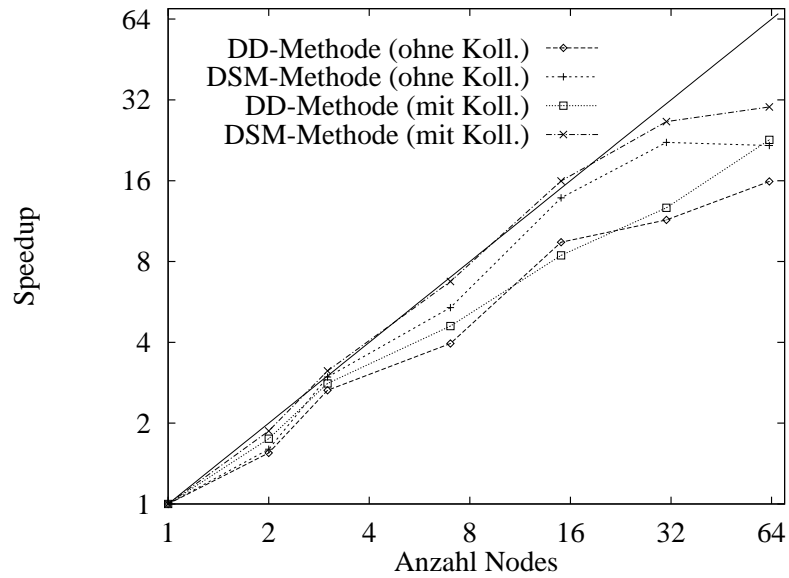


Abbildung 6.7: Speedup für die parallele Trajektorienberechnung: Vergleich der DD- und DSM-Methode für den kollisionsfreien und kollisionsbehafteten Fall

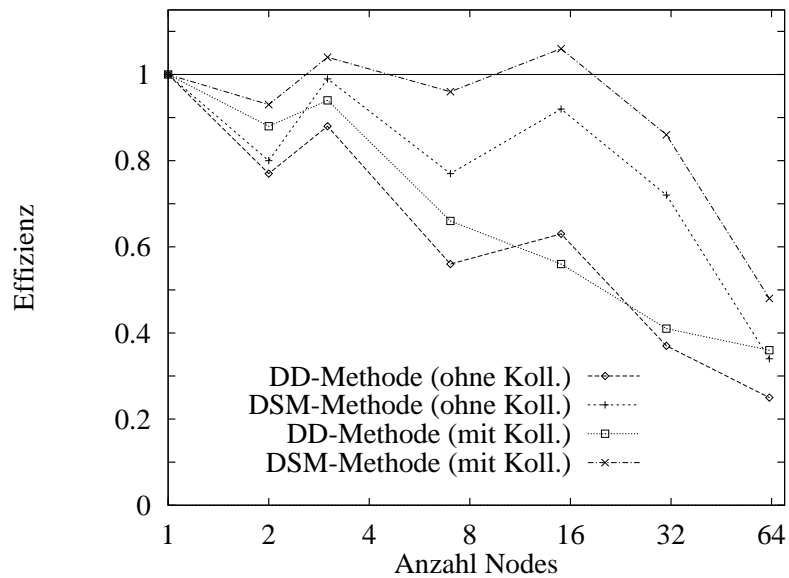


Abbildung 6.8: Effizienz für die parallele Trajektorienberechnung: Vergleich der DD- und DSM-Methode für den kollisionsfreien und kollisionsbehafteten Fall

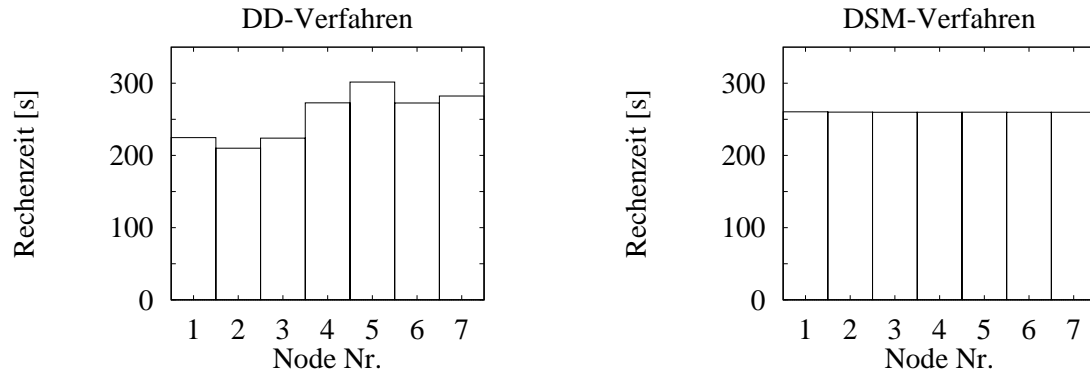


Abbildung 6.9: Verteilung der Rechenlast unter den Nodes (kollisionsfreier Fall, 7 Partitionen)

fizienzen erzielt werden. Dies ist dadurch begründet, daß der Rechenaufwand für die Trajektorienberechnung nicht gleichmäßig über die Länge des Kanals verteilt ist. Eine Folge davon ist beim DD-Verfahren eine ungleichmäßige Rechenlastverteilung unter den Nodes. Im Gegensatz dazu ist beim DSM-Verfahren durch die dynamische Zuweisung von Partitionen zu Nodes die Rechenlast stets optimal verteilt. Dies sei anhand der Abbildung 6.9 verdeutlicht. Hierin ist die Rechenzeit der einzelnen Nodes für den Testfall der kollisionsfreien Strömung in der 7-Block-Geometrie dargestellt. Dabei handelt es sich ausschließlich um Zeit, die für mathematische Operationen aufgewendet wird, d.h. der Zeitaufwand für den Datenaustausch ist darin nicht enthalten. Für das DD-Verfahren existiert eine feste Zuordnung von Gitterpartitionen zu Node-Prozessen, wobei die am Einströmrand gelegene Partition dem Node 1 und die am Ausströmrand gelegene dem Node 7 zugeordnet ist. In dem Diagramm für das DD-Verfahren ist zu sehen, daß der Rechenaufwand im oberen Teil des Kanals größer ist als im unteren Teil. Dies ist darauf zurückzuführen, daß die Partikel beim Eintritt in den Kanal zunächst nur eine vertikale Geschwindigkeitskomponente besitzen und daher die unteren Teilgebiete relativ schnell durchqueren. Auf ihrer Bahn werden die Partikel durch die Fluidturbulenz und die Saffman-Kraft in horizontaler Richtung abgelenkt, was im oberen Teil des Kanals zu häufigen Wandstößen führt. Diese verursachen wiederum eine Verstärkung der horizontalen Komponente der Partikelbewegung, was zur Folge hat, daß die Partikel sich in den oberen Teilgebieten länger aufhalten als in den unteren. Beim DSM-Verfahren wird dagegen die Rechenlast dynamisch verteilt, wodurch die Rechenzeiten der einzelnen Nodes nahezu identisch sind (siehe Abbildung 6.9, rechts).

Vergleicht man in den Abbildungen 6.7 und 6.8 die Ergebnisse für die kollisionsfreie und die kollisionsbehaftete Simulation, so sieht man, daß sowohl für das DD- als

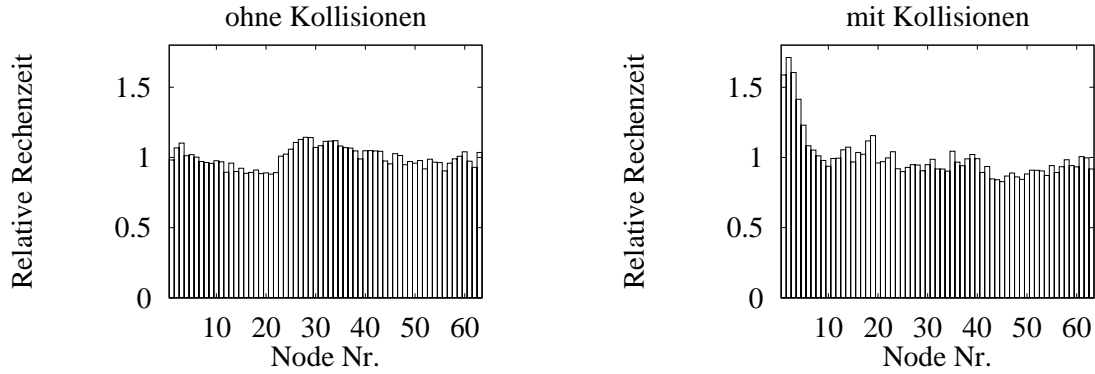


Abbildung 6.10: Verteilung des Rechenaufwandes über die Kanallänge

auch für das DSM-Verfahren fast durchweg bei der kollisionsbehafteten Simulation höhere Speedups bzw. Effizienzen erzielt werden. Lediglich bei der Anwendung der DD-Methode auf den 15-Block-Testfall schneidet die kollisionsfreie Berechnung besser ab. Die Höhe des erzielbaren Speedups bzw. der erzielbaren Effizienz ist von zwei wichtigen Faktoren abhängig: Zum einen von der Skalierbarkeit des seriellen Ausgangsproblems und zum anderen vom Verhältnis der Kommunikationszeit zur reinen Rechenzeit. In der Abbildung 6.10 sind für das DD-Verfahren und den 63-Block-Fall die relativen Rechenzeiten der einzelnen Nodes dargestellt. Hierbei handelt es sich wiederum nur um die Zeit, die für mathematische Berechnungen aufgewendet wurde, und die Kommunikationszeit ist darin nicht enthalten. Wie auch in der Abbildung 6.9 ist dem Node 1 die Partition zugeordnet, die am Einströmrand liegt. Der Node 63 bearbeitet das am Ausströmrand gelegene Teilgebiet. Anhand dieser Darstellung ist zu erkennen, daß sowohl im kollisionsfreien als auch im kollisionsbehafteten Fall der Rechenaufwand nicht gleichmäßig über die Länge des Kanals verteilt ist, sondern lokal variiert. Wenn nun der Kanal in Teilgebiete gleicher Größe aufgeteilt wird, hat dies zur Folge, daß der auf die einzelnen Gebiete entfallende Rechenaufwand im allgemeinen unterschiedlich groß ist. Das serielle Ausgangsproblem ist also durch diese Art der Partitionierung nicht optimal skalierbar.

Anhand der Tabelle 6.2 sei dieser Sachverhalt verdeutlicht. Hierin ist $T_{R,max}$ die maximale Rechenzeit (ohne Kommunikationszeit), die ein Node im DD-Verfahren zur Bearbeitung der ihm zugewiesenen Partition benötigte, T_1 ist die Rechenzeit für die serielle Berechnung der Kanalströmung und N die Anzahl der Partitionen bzw. Nodes. Die Größe $T_{R,max}$ hat einen großen Einfluß auf die Leistungsfähigkeit eines parallelen Verfahrens, da der Node mit der größten Rechenzeit in der Regel die Gesamtzeit der parallelen Berechnung bestimmt. In der zweiten und dritten Spalte der Tabelle 6.2 ist das Verhältnis von $T_{R,max}$ zum Quotienten T_1/N dargestellt.

N	$N \cdot T_{R,max}/T_1$		$T_{Komm}/T_{R,max}$	
	ohne Koll.	mit Koll.	ohne Koll.	mit Koll.
1	1	1	0	0
2	1.19	1.05	0.09	0.08
3	1.00	0.95	0.13	0.12
7	1.50	1.27	0.18	0.19
15	1.06	1.41	0.49	0.27
31	1.50	1.61	0.82	0.52
63	1.26	1.73	2.14	0.60

Tabelle 6.2: Skalierung des seriellen Ausgangsproblems und Verhältnis von Kommunikations- zu Rechenzeit

Dieses Verhältnis ist ein Maß für die Skalierung des Ausgangsproblems, die mit der gewählten Gitterpartitionierung erreicht wurde. Je kleiner dieser Wert ist, um so besser ist die Skalierung und um so größer ist der erreichbare Speedup bzw. die Effizienz. Der Vergleich zwischen den Werten für die kollisionsfreien und die kollisionsbehafteten Rechnungen zeigt, daß für kleine Partitionszahlen bis einschließlich sieben die kollisionsbehaftete Berechnung besser skaliert wird. Für die größeren Anzahlen von Partitionen wird im kollisionsfreien Fall eine bessere Skalierung erreicht. Dennoch sind der Speedup und die Effizienz im 31- und 63-Block-Fall für die kollisionsbehaftete Simulation höher als für die kollisionsfreie (siehe Abbildungen 6.7 und 6.8). Dies ist auf den Einfluß der Kommunikationszeit zurückzuführen. In der vierten und fünften Spalte der Tabelle ist jeweils das Verhältnis der Kommunikationszeit T_{Komm} zu $T_{R,max}$ angegeben. Hierbei sieht man, daß insbesondere bei den kollisionsfreien Rechnungen und großen Partitionszahlen der Anteil der Kommunikationszeit sehr stark zunimmt. Im selben Maße steigt die Gesamtzeit für die parallele Berechnung an, was sich ungünstig auf die Leistungsparameter des parallelen Verfahrens auswirkt. Im kollisionsbehafteten Fall nimmt die Kommunikationszeit ebenfalls zu, jedoch weniger stark. Dies liegt daran, daß bei den größeren Partitions- bzw. Node-Anzahlen die Kommunikationszeit hauptsächlich aus Wartezeit besteht, während der die Nodes auf die Zuweisung einer Partikelstartbedingung durch den Host warten. Diese Wartezeit ist im kollisionsbehafteten Fall kürzer, weil die Partikel aufgrund ihrer höheren mittleren Geschwindigkeit (siehe Abbildung 6.5) die

Partitionen schneller durchqueren.

Zusammenfassend läßt sich also feststellen, daß sowohl die Skalierbarkeit des seriellen Ausgangsproblems als auch die Kommunikationszeit einen großen Einfluß auf die Leistungsfähigkeit der parallelen Verfahren haben. Die oben für das Beispiel des DD-Verfahrens getroffenen Feststellungen zur Skalierbarkeit gelten prinzipiell auch für das DSM-Verfahren. Zwar wird beim DSM-Verfahren der Rechenaufwand insgesamt gleichmäßiger unter den Nodes aufgeteilt. Wenn jedoch die Teilprobleme, die den Nodes zugewiesen werden müssen, eine unterschiedliche endliche Größe besitzen, ist eine ideal gleichmäßige Rechenlastverteilung in der Regel nicht zu erreichen. Dieser Sachverhalt ist in der Abbildung 6.11 vereinfacht dargestellt. In diesem Beispiel wird eine Anzahl von Rechenaufgaben unterschiedlicher Größe statisch bzw. dynamisch auf drei Prozessoren verteilt. Man sieht, daß bei der dynamischen Verteilung die Rechenzeit insgesamt verkürzt wird. Die Rechenzeitunterschiede zwischen den Prozessoren werden zwar im dynamischen Fall verringert, sind jedoch weiterhin vorhanden. Aus diesem Grund wirken sich die oben beschriebenen Probleme der Skalierbarkeit auch auf die Leistungsfähigkeit des DSM-Verfahrens aus.

In den Abbildungen 6.7 bzw. 6.8 sind in zwei Fällen Speedups mit $Sp(N) > N$ bzw. Effizienzen zu beobachten, die größer als 1 sind. Dieser Effekt ist weder auf die Skalierung noch auf die Kommunikation zurückzuführen, sondern auf die Speicherorganisation der Prozessoren des Parallelrechners. Bei der parallelen Berechnung ist die Gesamtmenge der zu speichernden Daten wesentlich kleiner als im seriellen Fall. Dies hat zur Folge, daß ein signifikanter Anteil der Daten im Cache des Prozessors gespeichert werden kann. Wenn diese Daten für die Berechnung benötigt werden, stehen sie schneller zur Verfügung, als wenn sie zuerst vom Hauptspeicher zur CPU transportiert werden müßten. Dadurch wird die Ausführungszeit für die Rechnung verkürzt, was in einigen Fällen zum Erzielen eines superlinearen Speedups führen kann.

Parallele simultane Partikelverfolgung

Zur Untersuchung des parallelen Algorithmus für die simultane Partikelverfolgung wurden vier verschiedene Testfälle betrachtet. Zum einen wurden parallele Simulationen mit und ohne Berücksichtigung der Kollisionen durchgeführt, zum anderen wurde der Einfluß der Partikelverteilung im Strömungsgebiet auf die Leistungsfähigkeit des parallelen Verfahrens untersucht. Da die simultane Partikelverfolgung ein inhärent instationäres Verfahren ist, ändert sich diese Verteilung im allgemeinen im Verlaufe einer Simulation. Bei der Berechnung der vertikalen Kanalströmung waren die Partikel zu Beginn der Rechnung, d.h. zum Zeitpunkt $t = 0$ s gleichmäßig in der Geometrie verteilt. Dieser Zustand wird im folgenden als "Startverteilung" bezeichnet. Die Partikelverteilung, die dem stationären Strömungszustand entspricht und

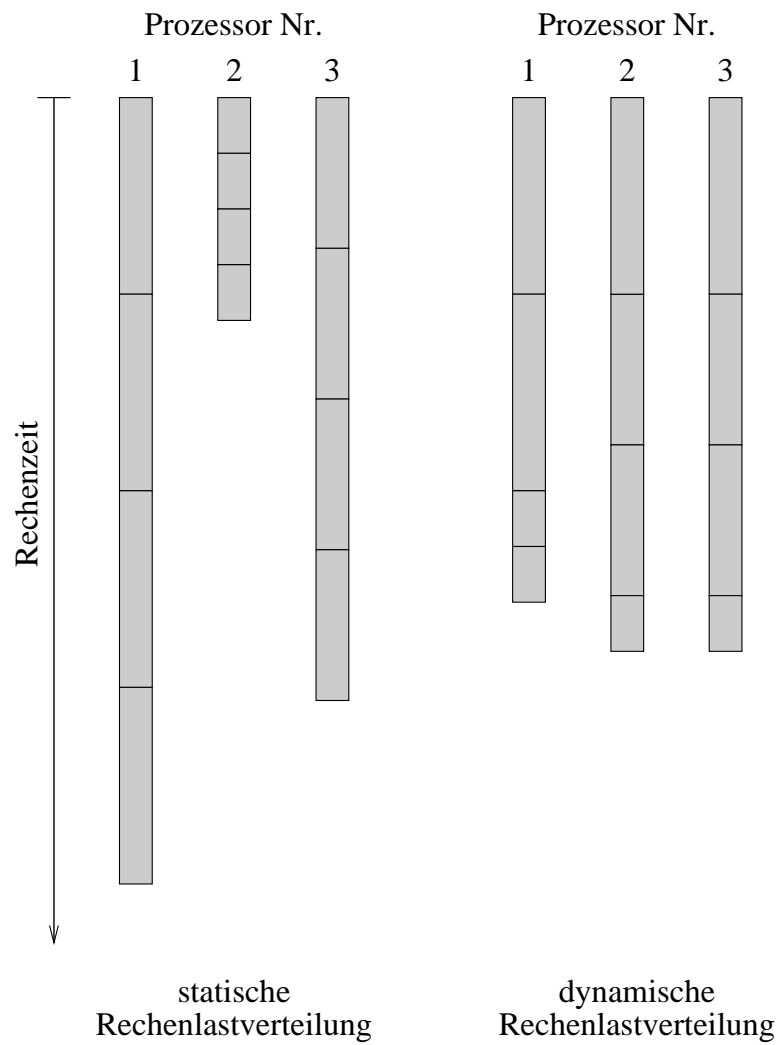


Abbildung 6.11: Vergleich zwischen der statischen und dynamischen Verteilung von Rechenaufgaben an drei Prozessoren

die sich erst nach einer bestimmten Zeit einstellt ($t = 0.6\text{ s}$ im kollisionsfreien bzw. $t = 0.56\text{ s}$ im kollisionsbehafteten Fall), soll als “Endverteilung“ bezeichnet werden. Zur Untersuchung der Leistungsfähigkeit des parallelen Algorithmus wurden Testrechnungen sowohl für die Start- als auch für die Endverteilung im kollisionsfreien und kollisionsbehafteten Fall durchgeführt. Die bei diesen Simulationen verwendeten Parameter sind in der Tabelle 6.3 zusammengefaßt.

Das Gitternetz für das Kanalsegment bestand im seriellen Fall aus 128×16 Gitterzellen. Für die parallelen Testrechnungen wurde das Gitter in 2, 4, 8, 16, 32 und 64 Partitionen aufgeteilt. Die Aufteilung wurde in der Weise durchgeführt, daß die einzelnen Partitionen jeweils dieselbe Anzahl von Gitterzellen enthielten und gleichzeitig die Anzahl von Zellen an den Partitions Grenzen minimal war. In der Abbildung 6.12 ist für drei der Testgitter jeweils ein Ausschnitt dargestellt. Der linke Teil der Abbildung zeigt das Gitter für den seriellen Fall. Für die Partitionsanzahlen bis einschließlich 16 wurde die Geometrie nur in vertikal übereinander liegende Segmente unterteilt, wie in der Mitte der Abbildung dargestellt. Für den 32-Block-Fall erfolgte erstmals ein Schnitt in vertikaler Richtung, so daß 16 Partitionen in Richtung der Kanalachse und zwei Partitionen in Querrichtung entstanden. Dieser Fall ist im rechten Teil der Abbildung 6.12 gezeigt. Zur nochmaligen Verdopplung der Anzahl der Partitionen für das 64-Block-Gitter wurde wiederum jedes Teilgebiet in zwei vertikal übereinander liegende Teilgebiete aufgeteilt.

Der Verlauf des Speedups bzw. der Effizienz des parallelen Algorithmus für die simultane Partikelverfolgung ist in den Abbildungen 6.13 bzw. 6.14 dargestellt. Hierin ist zu sehen, daß bis zu einer Prozessor- bzw. Partitionsanzahl von 16 in allen Fällen relativ gute Ergebnisse erzielt werden. Oberhalb dieser Anzahl sinkt jedoch die Leistungsfähigkeit des parallelen Verfahrens sehr stark ab. Der Grund hierfür ist die Art der Partitionierung der Gittergeometrie. Für die 2- bis 16-Block-Geometrien wird, wie oben beschrieben, das Kanalsegment in gleichgroße Teilgebiete entlang der Kanalachse aufgeteilt (siehe Abbildung 6.12, mitte). Dabei ist der auf die einzelnen Teilgebiete entfallende Rechenaufwand ungefähr gleich groß, da aufgrund der Verwendung periodischer Randbedingungen an den Kanalenden der Gesamtrechenaufwand relativ gleichmäßig über die Länge des Kanalsegmentes verteilt ist. Im 32- und 64-Block-Fall wird jedoch zusätzlich eine Teilung in vertikaler Richtung vorgenommen. Die dadurch entstehenden Teilgebiete enthalten zwar dieselbe Anzahl von Gitterzellen, haben jedoch einen unterschiedlichen Flächeninhalt, da die Gitterzellen zur Kanalwand hin kleiner werden (siehe Abbildung 6.12, rechts). Die größeren, an der Kanalachse gelegenen Partitionen enthalten mehr Partikel, so daß der Rechenaufwand in diesen Gebieten größer ist. Dadurch kommt es zu einer ungleichmäßigen Verteilung der Rechenlast unter den Prozessoren, was den Speedup bzw. die Effizienz des parallelen Algorithmus senkt.

Die Effizienzabnahme bei den 32- bzw. 64-Block-Rechnungen äußert sich im Fall

	ohne Kollisionen		mit Kollisionen	
	Startvert.	Endvert.	Startvert.	Endvert.
Anzahl Partikel	4 000	10 000	4 000	10 000
Anzahl Zeitschritte	750	220	700	220
Größe eines Zeitschritts Δt	10^{-4} s	10^{-5} s	10^{-4} s	10^{-5} s

Tabelle 6.3: Simulationsparameter für die parallele Berechnung der vertikalen Kanalströmung unter Verwendung der simultanen Partikelverfolgung

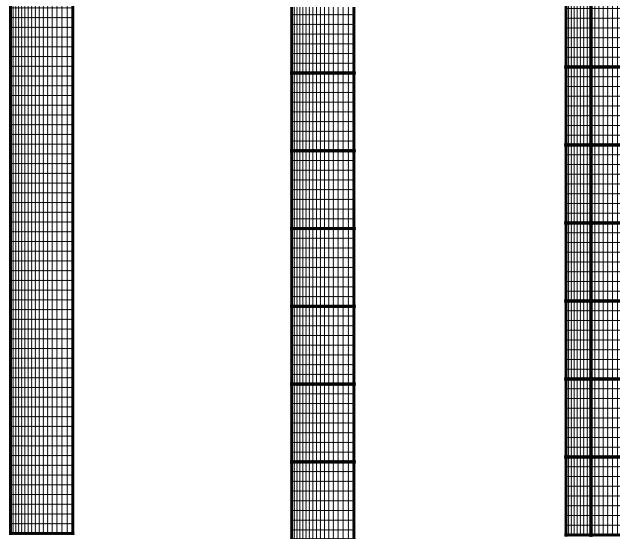


Abbildung 6.12: Prinzip der Gitterpartitionierung für die parallele simultane Partikelverfolgung: Gitter für die serielle Berechnung (links), 16-Block-Gitter (mitte) und 32-Block-Gitter (rechts)

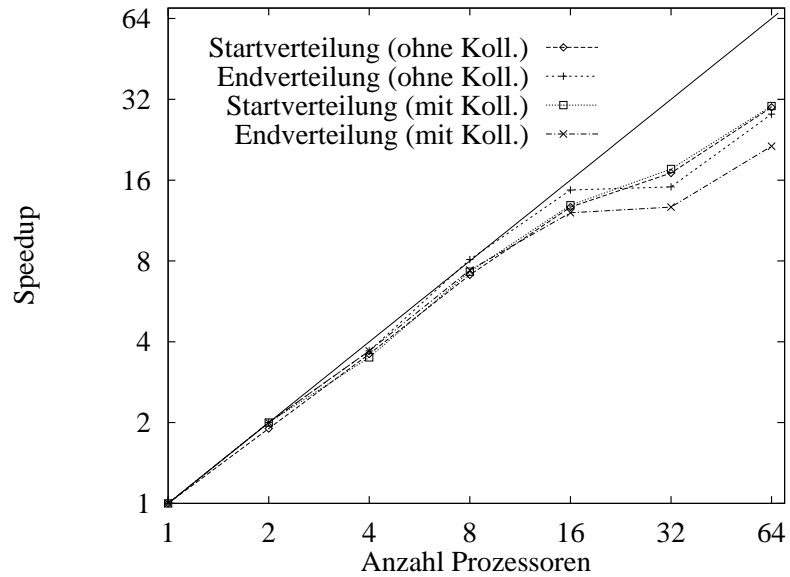


Abbildung 6.13: Speedup für die parallele Simulation auf der Basis der simultanen Partikelverfolgung

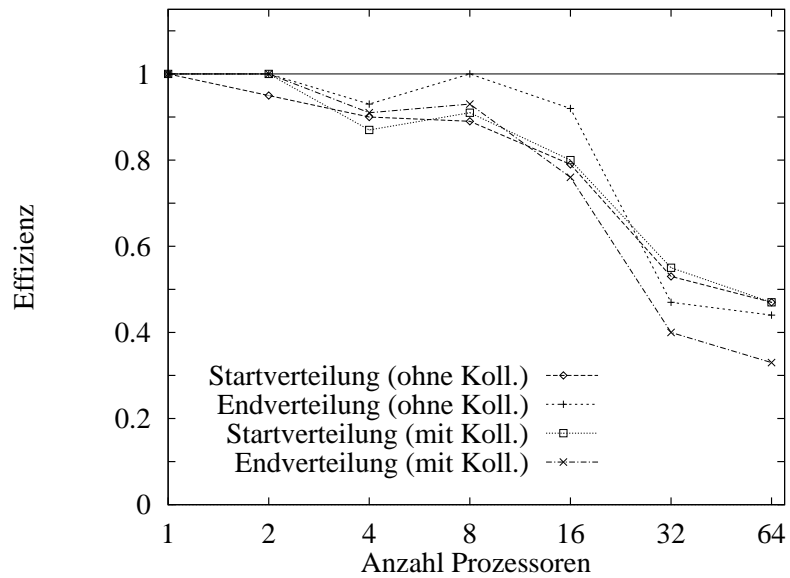


Abbildung 6.14: Effizienz für die parallelen Simulation auf der Basis der simultanen Partikelverfolgung

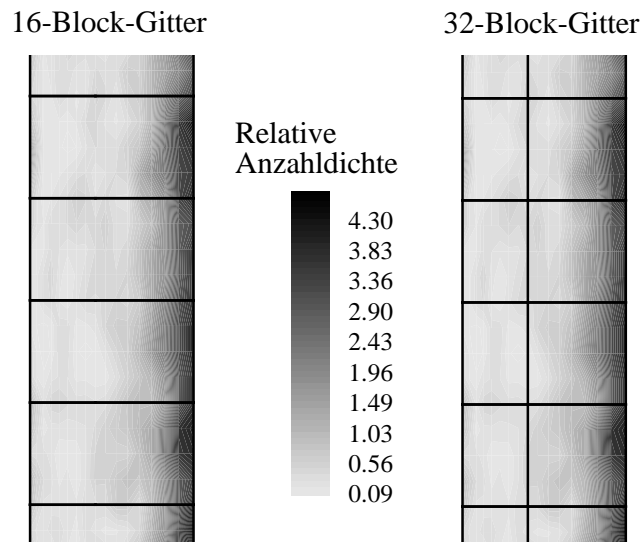


Abbildung 6.15: Verteilung der Partikel auf die Partitionen für den stationären Zustand im kollisionsfreien Fall

der Endverteilung wesentlich stärker als im Fall der Startverteilung. Dies ist dadurch begründet, daß bei der Endverteilung der Partikel, die dem stationären Strömungszustand entspricht, die Partikelkonzentration in der Nähe der Kanalachse wesentlich größer ist als in der Nähe der Wand (siehe Abbildungen 6.2 und 6.3). Im Vergleich zur Startverteilung ergibt sich daher ein wesentlich größeres Ungleichgewicht im Rechenaufwand zwischen den an der Kanalachse und den an der Kanalwand gelegenen Partitionen. In der Abbildung 6.15 ist für die kollisionsfreie Strömung die Verteilung der Partikelanzahldichte jeweils für einen Ausschnitt aus der 16- und der 32-Block-Geometrie dargestellt. Man kann erkennen, daß durch die Verdopplung der Anzahl der Partitionen der Rechenaufwand, der auf eine einzelne Partition entfällt, praktisch nicht verringert wird. Auch für die 32-Block-Geometrie ist nahezu der gesamte Rechenaufwand auf lediglich 16 Partitionen verteilt. Dies ist auch daran zu erkennen, daß bei den parallelen Testrechnungen im Fall der Endverteilung der Speedup für 32 und 64 Prozessoren nahezu identisch ist. Die Verdopplung der Anzahl der Prozessoren hat also in diesem Fall keine Verkürzung der Rechenzeit zur Folge.

Vergleicht man die Ergebnisse der kollisionsfreien und der kollisionsbehafteten Testrechnungen miteinander, so sieht man, daß im Fall der Endverteilung die kollisionsbehafteten Rechnungen niedrigere Effizienzen bzw. Speedups erzielen, obwohl in der

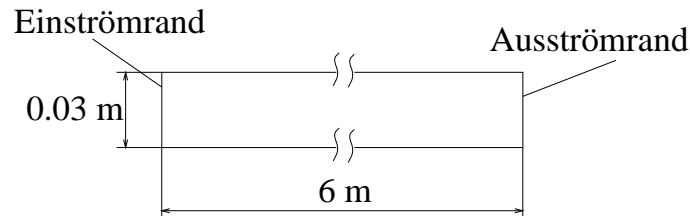


Abbildung 6.16: Strömungsgeometrie für die Berechnung der horizontalen Kanalströmung mit Hilfe der trajektorienbasierten Simulation

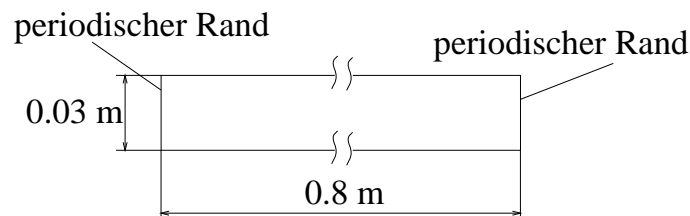


Abbildung 6.17: Strömungsgeometrie für die Berechnung der horizontalen Kanalströmung mit Hilfe der simultanen Partikelverfolgung

kollisionsbehafteten Strömung die Partikelverteilung gleichmäßiger ist. Dies ist auf einen höheren Kommunikationsaufwand zurückzuführen. In der kollisionsbehafteten Strömung ist die mittlere Partikelgeschwindigkeit größer als in der kollisionsfreien. Dies hat zur Folge, daß die Anzahl der Partikeln, die pro Zeiteinheit eine Partitions-grenze überqueren, größer ist als im kollisionsfreien Fall. Somit ist das Verhältnis der Kommunikations- zur Rechenzeit im kollisionsbehafteten Fall ungünstiger.

6.2 Berechnung der Strömung in einem horizontalen Kanal

6.2.1 Physikalische Eigenschaften der Strömung

Als zweites Testbeispiel wurde eine Gas-Partikel-Strömung in einem horizontalen Kanal untersucht. Für die trajektorienbasierte Simulation hatte das Strömungsgebiet eine Länge von 6 m und eine Höhe von 0.03 m. Für die Strömungsberechnung mit Hilfe der simultanen Partikelverfolgung wurde ein 0.8 m langes Kanalsegment mit periodischen Randbedingungen an den Enden des Segmentes verwendet. In den

Abbildung 6.16 und 6.17 sind die Abmessungen dieser beiden Geometrien schematisch dargestellt.

Von Oesterlé und Petitjean [34] wurden Simulationen für horizontale Rohrströmungen bei verschiedenen Partikelbeladungen durchgeführt. Um zumindest einen qualitativen Vergleich der im Rahmen der vorliegenden Arbeit durchgeführten Berechnungen mit den Simulationen von Oesterlé und Petitjean [34] zu ermöglichen, wurden die dort verwendeten physikalischen Strömungsparameter hier weitgehend übernommen. Als Gasphase wurde Luft verwendet, bei der dispersen Phase handelte es sich um Glaspartikel, und die Kanal- bzw. Rohrwand bestand aus Kupfer. In der Tabelle 6.4 sind die Werte der wichtigsten physikalischen Parameter zusammengefaßt.

Wie bei der Berechnung der vertikalen Kanalströmung wurde auch hier angenommen, daß die Strömungsgrößen der Gasphase über die Länge des Kanals konstant sind. Des weiteren wurde aus den gleichen Gründen, die bereits im Abschnitt 6.1.1 erläutert wurden, der Einfluß der Partikel auf die Bewegung des Gases nicht berücksichtigt.

Es wurde wiederum jeweils eine Strömung mit einer niedrigen Beladung ($\alpha = 4.62 \cdot 10^{-6}$, $\eta = 0.01$) und einer hohen Beladung ($\alpha = 4.62 \cdot 10^{-3}$, $\eta = 10$) untersucht. Die Berechnung der ersteren erfolgte unter Vernachlässigung der Partikel-Partikel-Kollisionen. Beide Strömungen wurden sowohl mit Hilfe der Methode der Trajektorienberechnung als auch mit der simultanen Partikelverfolgung berechnet. Die hierbei verwendeten Simulationsparameter sind im folgenden zusammengefaßt:

Trajektorienberechnung

1. Kollisionsfreie Strömung: Es wurden 5 000 Trajektorien vom Einström- bis zum Ausströmrand berechnet. Wie auch im Fall der vertikalen Kanalströmung ist diese Anzahl aufgrund der relativ einfachen Strömungsgeometrie groß genug, um statistisch gesicherte Werte für die gemittelten Strömungsgrößen der Partikelphase zu erhalten.
2. Kollisionsbehaftete Strömung: Auch hierbei wurden 5 000 Trajektorien berechnet sowie zehn aufeinanderfolgende Trajektorienberechnungen zur Berücksichtigung der Kollisionen ausgeführt. Ähnlich wie bei der Simulation der vertikalen Kanalströmung zeigte sich bei einer Erhöhung der Anzahl der Trajektorien bzw. Iterationen keine Veränderung der Simulationsergebnisse.

Simultane Partikelverfolgung

1. Kollisionsfreie Strömung: Hierbei wurde die Bewegung von 10 000 Partikeln berechnet. Aufgrund der Verwendung von periodischen Randbedingungen an

Physikalische Größe	Formel- zeichen	Wert
Kanalhöhe	H	30 mm
Mittlere axiale Gasgeschwindigkeit	\bar{v}_F	25.5 m/s
Dichte des Gases	ρ_F	1.21 kg/m ³
Dynamische Viskosität des Gases	μ	$1.79 \cdot 10^{-5}$ kg/(m s)
Mittlerer Partikeldurchmesser	\bar{d}_P	100 μ m
Materialdichte der Partikel	ρ_P	2620 kg/m ³
Stoßzahl für Part.–Wand–Stoß	e_w	0.9
Reibungsbeiwert für Part.–Wand–Stoß	f_w	0.53
Stoßzahl für Part.–Part.–Stoß	e_p	0.95
Reibungsbeiwert für Part.–Part.–Stoß	f_p	0.4
Volumetrische Beladung	α_P	<div>ohne Koll.: $4.62 \cdot 10^{-6}$</div> <hr/> <div>mit Koll.: $4.62 \cdot 10^{-3}$</div>
Massenbeladung	η	<div>ohne Koll.: 0.01</div> <hr/> <div>mit Koll.: 10</div>

Tabelle 6.4: Physikalische Parameter für die Simulation der horizontalen Kanalströmung

den Enden des Kanalsegmentes war diese Anzahl während der gesamten Simulation konstant. Die physikalische Laufzeit der Simulation betrug 0.3 s, wobei der minimal zugelassene Zeitschritt $\Delta t_{min} = 10^{-5}$ s betrug.

2. Kollisionsbehaftete Strömung: Auch hier wurden 10 000 Partikel, eine Laufzeit von 0.3 s und ein minimaler Zeitschritt von $\Delta t_{min} = 10^{-5}$ s verwendet. Die Partikel–Partikel–Kollisionen wurden wiederum mit Hilfe der Monte–Carlo–Methode berechnet.

In den Abbildungen 6.18 bis 6.21 sind die physikalischen Ergebnisse für die Simulation der horizontalen Kanalströmung dargestellt. In diesen Diagrammen bezeichnet die auf der vertikalen Koordinatenachse aufgetragene Größe y jeweils die Höhe über dem Kanalboden. Die Abbildung 6.18 zeigt die relative Anzahldichte der Partikel für die kollisionsfreie Strömung in einer Entfernung von 6 m vom Einstörmrand. Die Anzahldichte ist relativ gleichförmig über die Kanalhöhe verteilt. Der Grund hierfür ist, daß die Partikel durch die Wirkung der Schwerkraft häufig mit dem Boden des Kanals kollidieren, von wo sie wegen der Rauigkeit der Wand stark resuspendiert werden. In einer dreidimensionalen Rohrströmung ist dieser Resuspensionseffekt schwächer, da die Partikelbewegung eine zusätzliche horizontale Komponente besitzt. Bei den Simulationen von Oesterlé und Petitjean [34] zeigt sich daher in der unteren Hälfte des Strömungsgebietes eine höhere Partikelkonzentration und der oberen Hälfte eine niedrigere als bei den Rechnungen der vorliegenden Arbeit.

In der Abbildung 6.19 ist die relative Anzahldichte für die kollisionsbehaftete Strömung gezeigt. Man sieht, daß die Ergebnisse der verschiedenen Berechnungen sich zwar quantitativ unterscheiden, jedoch zeigt sich in allen Fällen eine sehr ungleichförmige Partikelverteilung über der Kanalhöhe. Die Konzentration in der oberen Hälfte des Kanals ist deutlich niedriger als im kollisionsfreien Fall, wogegen die Konzentration in der unteren Hälfte deutlich höher ist. Dies ist damit begründet, daß die Partikel–Partikel–Kollisionen die durch die Wandstöße hervorgerufene Resuspension dämpfen. Ein Partikel, das sich nach einem Stoß mit dem Kanalboden in das Innere des Strömungsgebietes zurückbewegt, erreicht eine geringere Höhe als in einer verdünnten Strömung, da es auf seiner Bahn mit anderen Partikeln kollidiert.

Des weiteren ist anhand der Abbildung 6.19 zu erkennen, daß die maximale Partikelkonzentration nicht unmittelbar am Kanalboden auftritt, sondern in einem bestimmten Abstand von diesem. In diesem Bereich finden besonders viele Partikel–Partikel–Kollisionen statt, da hier die von der Wand zurückprallenden und die sich zur Wand hin bewegenden Partikel aufeinandertreffen. Durch die Kollisionen wird die vertikale Bewegung der zurückprallenden Partikel gedämpft. Umgekehrt wird die Bewegung der Partikel, die sich zur Wand hin bewegen, durch die zurückprallenden Partikel behindert. Als Folge davon steigt die Konzentration in diesem Gebiet stark an.

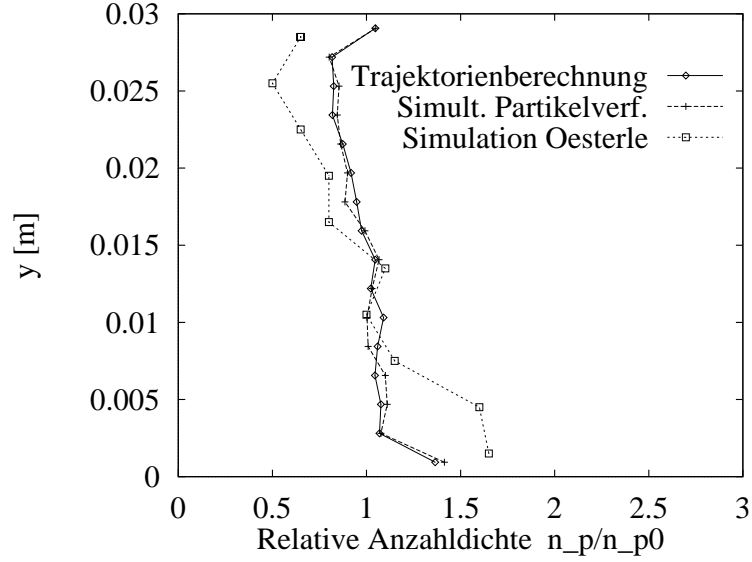


Abbildung 6.18: Relative Partikelanzahldichte n_P/n_{P0} für die kollisionsfreie Strömung ($\alpha_P = 4.62 \cdot 10^{-6}$, $\eta = 0.01$), $L = 6$ m stromab vom Kanal- bzw. Rohreinlauf

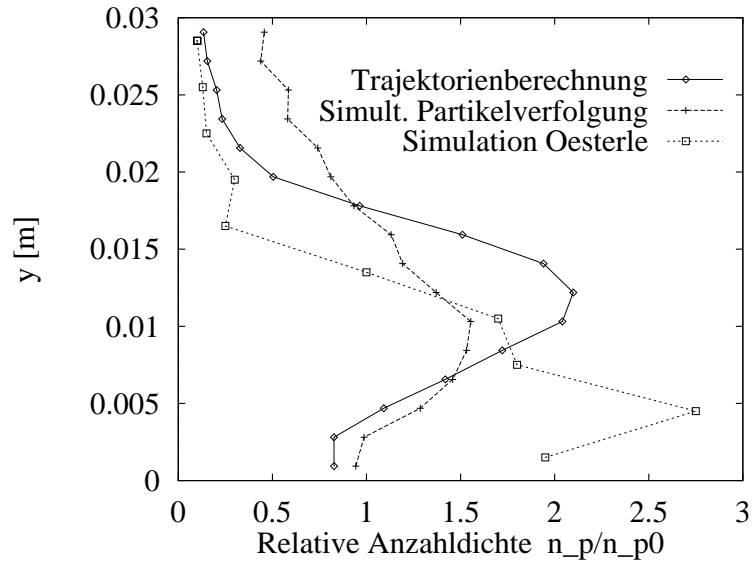


Abbildung 6.19: Relative Partikelanzahldichte n_P/n_{P0} für die kollisionsbehaftete Strömung ($\alpha_P = 4.62 \cdot 10^{-3}$, $\eta = 10.0$), $L = 6$ m stromab vom Kanal- bzw. Rohreinlauf

Bei der Simulation von Oesterlé und Petitjean [34] ist die maximale Anzahldichte größer als bei den Simulationen dieser Arbeit. Der Ort der maximalen Anzahldichte befindet sich außerdem in einem kleineren Abstand zum Kanalboden. Dies ist, ähnlich wie im kollisionsfreien Fall, darauf zurückzuführen, daß von Oesterlé und Petitjean [34] eine Rohrströmung betrachtet wurde, in der die Partikel weniger stark in vertikaler Richtung resuspendiert werden.

Der Vergleich der Ergebnisse der vorliegenden Arbeit für die Trajektorienberechnung und die simultane Partikelverfolgung im kollisionsbehafteten Fall zeigt, daß die maximale Anzahldichte, die mit Hilfe der simultanen Partikelverfolgung bestimmt wurde, kleiner ist als die durch die Trajektorienberechnung bestimmte. Ein möglicher Grund hierfür ist, daß für die Strömungsberechnung mit Hilfe der simultanen Partikelverfolgung eine konstante Anzahl von Partikeln und periodische Randbedingungen an den Kanalenden gewählt wurden. Dadurch ist die mittlere Partikelanzahldichte in der Strömungsgeometrie während der gesamten Rechnung konstant. Im Fall der Trajektorienberechnung ändert sich jedoch die mittlere Anzahldichte über die Länge des Kanals, wenn sich die mittlere Partikelgeschwindigkeit ändert. Da die Kollisionsfrequenz u.a. von der Anzahldichte abhängig ist, kann bei unterschiedlichen Anzahldichten der Einfluß der Kollisionen auf die makroskopischen Strömungsgrößen unterschiedlich sein. Auf eine genauere Untersuchung der Ursachen für die Differenzen zwischen den Ergebnissen der beiden Simulationsmethoden wurde hier verzichtet. Für solche detaillierten Untersuchungen sind die theoretischen Grundlagen der Kollisionsmodelle sowie die aus der Literatur bekannten praktischen Erfahrungen mit diesen Modellen bisher zu gering. Da der Schwerpunkt dieser Arbeit auf der Entwicklung und dem Test von parallelen Simulationsalgorithmen liegt, wird es hier als ausreichend angesehen, daß beide Berechnungsmethoden qualitativ ähnliche Ergebnisse liefern.

In den Abbildungen 6.20 und 6.21 ist der Verlauf der mittleren axialen Geschwindigkeit der Partikel- und Gasphase über die Kanalhöhe dargestellt. Für die Partikelgeschwindigkeit sind von Oesterlé und Petitjean [34] keine Ergebnisse veröffentlicht, so daß in diesen Diagrammen nur die Ergebnisse der Simulationen dieser Arbeit enthalten sind. Im kollisionsfreien Fall, der in der Abbildung 6.20 gezeigt ist, ist das Geschwindigkeitsprofil der Partikelphase relativ gleichförmig. Die mittlere Partikelgeschwindigkeit ist deutlich geringer als die mittlere Gasgeschwindigkeit. Die Partikel stoßen sehr häufig gegen die Kanalwände, wobei ihre kinetische Energie aufgrund der Wandreibung jedesmal verringert wird. Des weiteren ist die Zeit zwischen zwei Wandstößen relativ kurz, so daß die Partikel in dieser Zeit durch die aerodynamischen Kräfte des Gases nicht signifikant beschleunigt werden. Im Gegensatz dazu ist im kollisionsbehafteten Fall, der in der Abbildung 6.21 dargestellt ist, die mittlere Partikelgeschwindigkeit deutlich höher. Dies liegt daran, daß Partikel, die von der Wand zurückprallen, die Wand gegen eintreffende Partikel abschirmen.

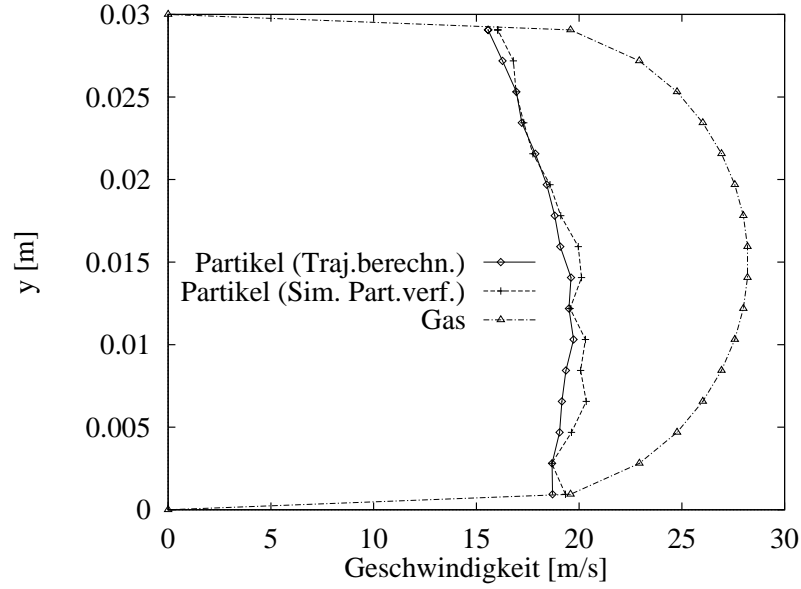


Abbildung 6.20: Mittlere axiale Geschwindigkeit für die kollisionsfreie Strömung ($\alpha_P = 4.62 \cdot 10^{-6}$, $\eta = 0.01$), $L = 6$ m stromab vom Kanaleinlauf

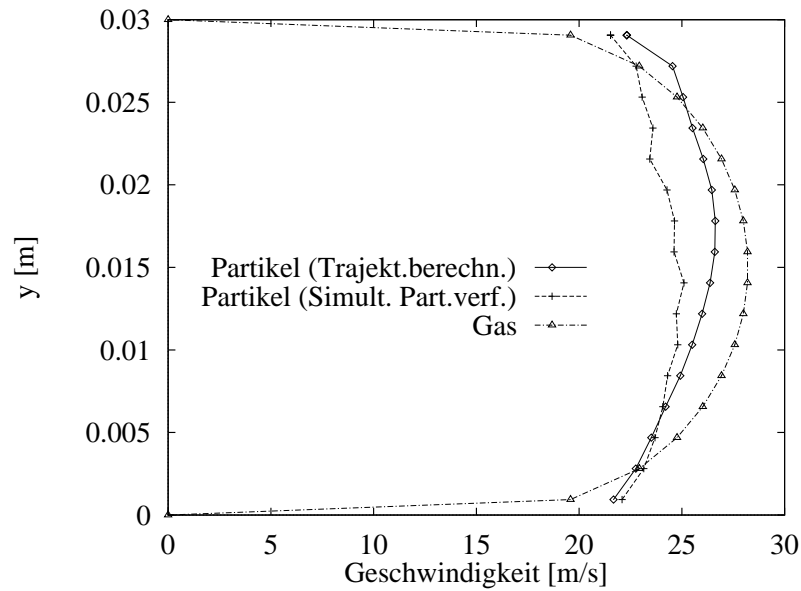


Abbildung 6.21: Mittlere axiale Geschwindigkeit für die kollisionsbehaftete Strömung ($\alpha_P = 4.62 \cdot 10^{-3}$, $\eta = 10.0$), $L = 6$ m stromab vom Kanaleinlauf

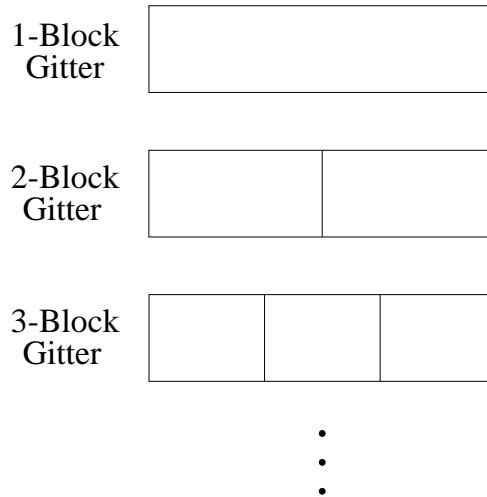


Abbildung 6.22: Aufteilung des Gitters für den horizontalen Kanal in Partitionen für die parallele Trajektorienberechnung

Als Folge davon stoßen die Partikel insgesamt seltener gegen die Kanalwände. Sie halten sich daher zwischen zwei Wandstößen länger in der Gasströmung auf und werden auf eine höhere Geschwindigkeit beschleunigt.

6.2.2 Anwendung der parallelen Simulationsalgorithmen auf die Berechnung der horizontalen Kanalströmung

Parallele Trajektorienberechnung

Zur Untersuchung der parallelen Verfahren für die Trajektorienberechnung wurde jeweils ein kollisionsfreier und ein kollisionsbehafteter Testfall betrachtet. Dabei wurden die folgenden Simulationsparameter verwendet:

1. Kollisionsfreier Testfall: In diesem Fall wurden 500 Trajektorien vom Einström- bis zum Ausströmrand berechnet und die dafür benötigte Rechenzeit bestimmt.
2. Kollisionsbehafteter Testfall: Hierbei wurden 200 Trajektorien berechnet. Wie bei der vertikalen Kanalströmung wurden auch hier zwei Iterationen des ITB-Algorithmus ausgeführt. Die Rechenzeit, die für die zweite, kollisionsbehaftete Iteration benötigt wurde, bildet die Vergleichsgröße zur Beurteilung der parallelen Verfahren.

Für die seriellen Referenzrechnungen bestand das Gitternetz für die in der Abbildung 6.16 dargestellte Strömungsgeometrie aus 1024×16 Gitterzellen. Dieses Gitternetz wurde für die parallelen Testrechnungen in 2, 3, 7, 15, 31 bzw. 63 Partitionen aufgeteilt. Die Gründe für die Wahl dieser Partitionsanzahlen sind dieselben, die bereits im Abschnitt 6.1.2 erläutert wurden. Die Aufteilung des Gitternetzes wurde wiederum in der Weise durchgeführt, daß die einzelnen Teilgitter dieselbe Anzahl von Gitterzellen enthielten sowie die Anzahl von Zellen an den Partitionsrändern minimiert wurde. Die Anwendung dieser Strategie hatte zur Folge, daß das Gitter in allen Fällen in horizontal nebeneinander liegende Segmente unterteilt wurde, wie in der Abbildung 6.22 schematisch gezeigt.

In den Abbildungen 6.23 und 6.24 sind der Speedup bzw. die Effizienz der parallelen Simulationsalgorithmen für die Trajektorienberechnung dargestellt. Wie auch im Fall der vertikalen Kanalströmung wurden bei der Bestimmung dieser Größen ausschließlich die Rechenzeiten der Node-Prozesse verwendet, da diese die eigentliche Berechnung der Trajektorien durchführen. Vergleicht man die Ergebnisse des DD- und des DSM-Verfahrens, so sieht man, daß mit Hilfe des DSM-Verfahrens höhere Speedups bzw. Effizienzen erzielt werden. Dies ist wiederum darauf zurückzuführen, daß bei der Verwendung dieses Verfahrens die Rechenlast wesentlich gleichmäßiger unter den Nodes verteilt ist als beim DD-Verfahren.

Der Vergleich zwischen Ergebnissen der kollisionsfreien und der kollisionsbehafteten Testrechnungen zeigt, daß für die Partitions- bzw. Node-Anzahlen 2 und 3 die kollisionsbehafteten Rechnungen Speedups und Effizienzen erzielen, die leicht über denen für die kollisionsfreien Rechnungen liegen. Im 8-Node-Fall sind die Werte nahezu gleich, und bei den größeren Node-Anzahlen erreichen die kollisionsfreien Testrechnungen bessere Ergebnisse. Die Erklärung dieses Verhaltens liefert wiederum die Betrachtung der Skalierbarkeit des seriellen Ausgangsproblems sowie des Anteils der Kommunikationszeit an der Gesamtrechenzeit. In der Abbildung 6.25 sind für den Fall des DD-Verfahrens und die 63-Block-Geometrie jeweils die relativen Rechenzeiten der einzelnen Nodes dargestellt. Dem Node 1 ist hierbei das unmittelbar am Einströmrand gelegene Teilgebiet zur Berechnung zugeordnet, und der Node 63 bearbeitet die am Ausströmrand gelegene Partition. Die Darstellungen in der Abbildung 6.25 geben daher einen Eindruck von der Verteilung des Rechenaufwandes über die Länge des Kanals. Diese Verteilung ist sowohl im kollisionsfreien als auch im kollisionsbehafteten Fall ungleichmäßig. Unmittelbar am Einströmrand ist ein Maximum im Rechenaufwand zu beobachten. Dies liegt daran, daß die Startgeschwindigkeit der Partikel am Einströmrand $v_p = 0$ m/s betrug. Die Partikel werden erst allmählich durch die Gasströmung beschleunigt und halten sich daher relativ lange in der ersten Partition auf. Sie werden zunächst horizontal beschleunigt, ohne gegen die Kanalwände oder gegen andere Partikel zu stoßen, weshalb der Rechenaufwand stromab vom Einströmrand stark absinkt. Erst nach einer bestimm-

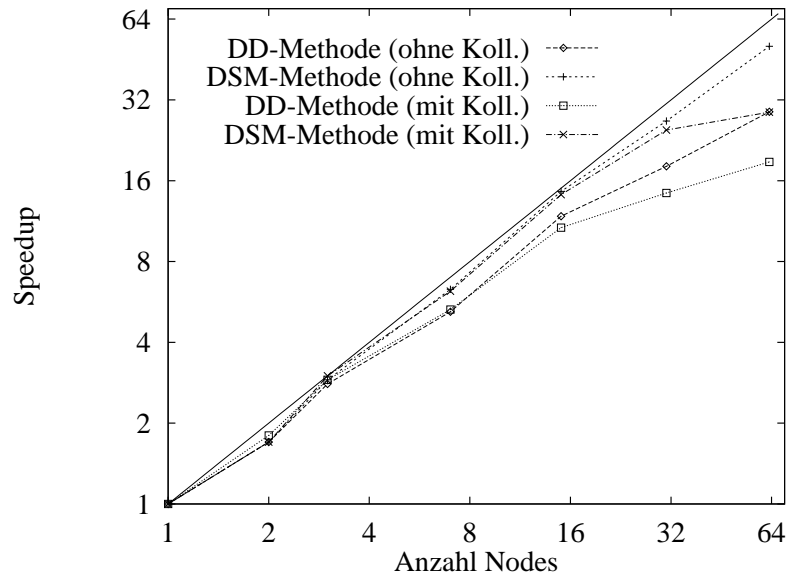


Abbildung 6.23: Speedup für die parallele Trajektorienberechnung: Vergleich der DD- und DSM-Methode für den kollisionsfreien und kollisionsbehafteten Fall

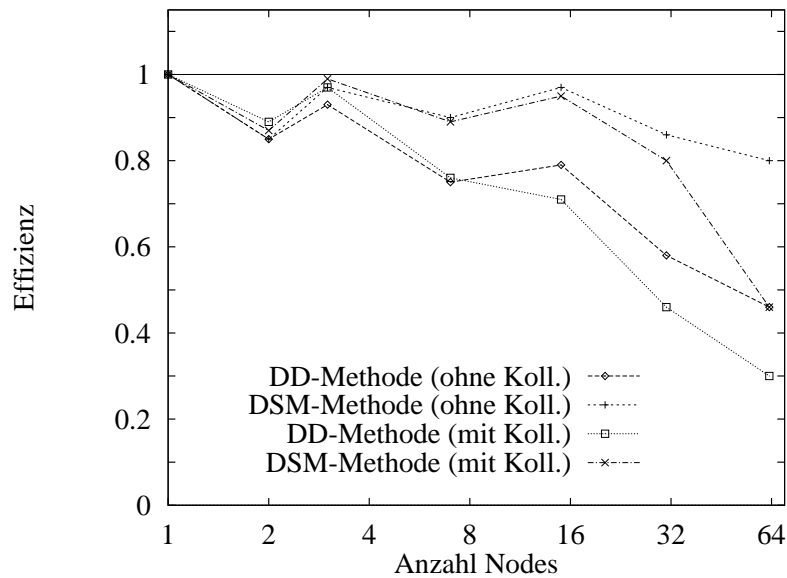


Abbildung 6.24: Effizienz für die parallele Trajektorienberechnung: Vergleich der DD- und DSM-Methode für den kollisionsfreien und kollisionsbehafteten Fall

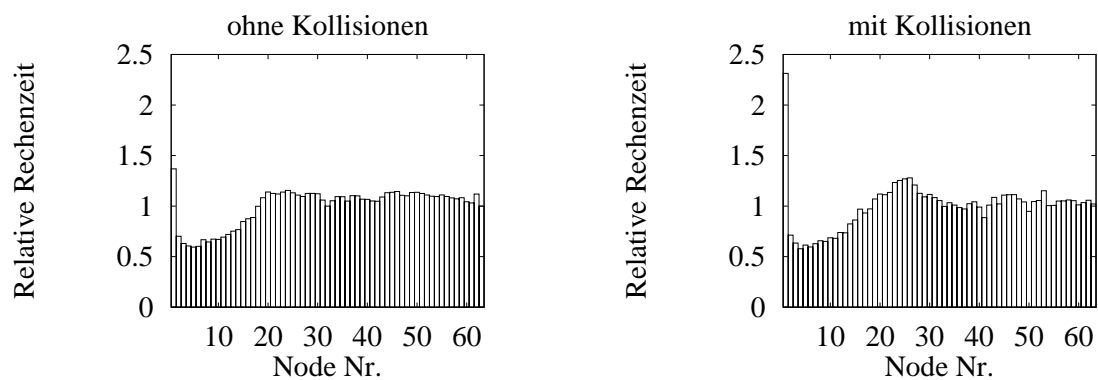


Abbildung 6.25: Verteilung des Rechenaufwandes über die Kanallänge

N	$N \cdot T_{R,max}/T_1$		$T_{Komm}/T_{R,max}$	
	ohne Koll.	mit Koll.	ohne Koll.	mit Koll.
1	1	1	0	0
2	1.18	1.09	0.002	0.03
3	1.06	1.01	0.01	0.02
7	1.20	1.22	0.11	0.08
15	1.10	1.08	0.15	0.29
31	1.24	1.46	0.38	0.49
63	1.55	2.65	0.41	0.27

Tabelle 6.5: Skalierung des seriellen Ausgangsproblems und Verhältnis von Kommunikations- zu Rechenzeit

ten Lauflänge nimmt die Anzahl der Partikel–Wand– und Partikel–Partikel–Stöße zu, und der Rechenaufwand steigt wieder an.

Da der Rechenaufwand ungleichmäßig über die Länge des Kanals verteilt ist, das Strömungsgebiet jedoch in Partitionen gleicher Größe entlang der Kanalachse aufgeteilt wird, ist die Rechenlast, die auf die einzelnen Partitionen entfällt, im allgemeinen unterschiedlich groß. Je nach Anzahl der Teilgebiete können dabei die Rechenlastunterschiede zwischen den Teilgebieten mehr oder weniger groß sein, d.h. das serielle Ausgangsproblem kann mehr oder weniger gut skaliert sein. Dies sei anhand der Tabelle 6.5 verdeutlicht. Hierin ist wiederum $T_{R,max}$ die maximale Rechenzeit, die ein Node bei der Anwendung des DD–Verfahrens zur Berechnung seines Teilproblems benötigte. T_1 ist die Rechenzeit für die serielle Rechnung, und das Verhältnis von $T_{R,max}$ zu T_1/N ist ein Maß dafür, wie gut das serielle Problem bei der Aufteilung in N Partitionen skaliert wurde. Je kleiner dieses Verhältnis ist, desto besser ist die Skalierung. Die Größe dieses Verhältnisses ist jeweils in der zweiten und dritten Spalte der Tabelle 6.5 angegeben. Außerdem enthält die vierte bzw. fünfte Spalte der Tabelle jeweils das Verhältnis der Kommunikationszeit T_{Komm} zu $T_{R,max}$. Für den 2– und 3–Block–Fall ist der Anteil der Kommunikationszeit jeweils sehr klein, und die Skalierung ist bei den kollisionsbehafteten Rechnungen günstiger, was eine höhere Effizienz zur Folge hat. Im 7–Block–Fall ist für die kollisionsfreie Rechnung eine bessere Skalierung zu beobachten als für die kollisionsbehaftete, jedoch ist der Anteil der Kommunikationszeit größer. Diese beiden Effekte heben sich gegenseitig auf, so daß die Effizienzen für diese beiden Rechnungen nahezu gleich sind. Wie oben bereits erwähnt, erzielen bei den größeren Blockanzahlen die kollisionsfreien Berechnungen jeweils die größeren Effizienzen. Aus der Tabelle 6.5 ist zu entnehmen, daß die Ursachen hierfür unterschiedlich sind. Im 15–Block–Fall liegt es an dem wesentlich kleineren Anteil der Kommunikationszeit, im 63–Block–Fall ist die Skalierung wesentlich besser, und für den 31–Block–Fall trifft beides zu.

Die in der Tabelle 6.5 angegebenen Werte gelten für das Beispiel des DD–Verfahrens. Wie jedoch im Abschnitt 6.1.2 erläutert wurde, wirken sich die Probleme der Skalierbarkeit auch auf die Leistungsfähigkeit des DSM–Verfahrens aus.

Es ist festzustellen, daß sowohl die Skalierung des Ausgangsproblems als auch der Anteil der Kommunikationszeit an der Gesamtausführungszeit große Auswirkungen auf die Leistungsfähigkeit der hier verwendeten parallelen Verfahren zur Trajektorienberechnung haben. Die oben dargestellten Beispiele zeigen, daß die Größe des Einflusses dieser beiden Faktoren von Fall zu Fall stark variieren kann.

Parallele simultane Partikelverfolgung

Wie im Fall der vertikalen Kanalströmung wurden auch hier vier verschiedene Testfälle betrachtet, um den parallelen Algorithmus für die simultane Partikelverfolgung zu untersuchen. Zum einen wurden Testrechnungen mit und ohne Berücksichtigung der Kollisionen durchgeführt. Zum anderen wurde wiederum der instationäre Charakter dieses Berechnungsverfahrens berücksichtigt. Dazu wurden Testrechnungen sowohl für den Zustand der Partikelphase zu Beginn einer Simulation, d.h. für die "Startverteilung" der Partikel, als auch für den stationären Zustand, d.h. die "Endverteilung", durchgeführt. Die in den einzelnen Testrechnungen verwendeten Simulationsparameter sind in der Tabelle 6.6 angegeben.

Für die serielle Referenzrechnung wurde das Kanalsegment (siehe Abbildung 6.17) mit 128×16 Gitterzellen vernetzt. Dieses Gitternetz wurde zur Untersuchung des parallelen Algorithmus in 2, 4, 8, 16, 32, 64 und 128 Partitionen aufgeteilt. Im 2- bis 16-Block-Fall wurde das Gitter ausschließlich in horizontal nebeneinander liegende Segmente unterteilt. Die Art der Partitionierung für die 32- bis 128-Block-Gitter ist in der Abbildung 6.26 dargestellt. Im Gegensatz zum Gitter für den vertikalen Kanal waren hier alle Gitterzellen gleich groß. Dies bedeutet, daß alle Partitionen eines jeweiligen Gitternetzes nicht nur dieselbe Anzahl von Gitterzellen enthielten, sondern auch dieselbe physikalische Größe besaßen.

In den Abbildungen 6.27 und 6.28 ist der Speedup bzw. die Effizienz für die parallele Simulation auf der Basis der simultanen Partikelverfolgung dargestellt. Für die kollisionsfreien Testrechnungen sowie für die kollisionsbehafteten Rechnungen im Fall der Startverteilung sind sowohl die Größe als auch der Verlauf des Speedups und der Effizienz nahezu gleich. In allen diesen Testfällen ist die Verteilung der Partikel im Strömungsgebiet relativ gleichmäßig. Die Startverteilung entspricht jeweils einer gleichmäßigen Zufallsverteilung der Partikel. Wie außerdem bereits anhand der Abbildung 6.18 gezeigt wurde, ist im kollisionsfreien Fall auch die Konzentrationsverteilung im stationären Endzustand relativ gleichförmig. In diesen Fällen ist also jeweils der Rechenaufwand, der auf die einzelnen Partitionen des Strömungsgebietes entfällt, ungefähr gleich groß. Die Abnahme der Effizienz mit wachsenden Partitions- bzw. Prozessor-Anzahlen ist im wesentlichen darauf zurückzuführen, daß die Größe der Partitionen abnimmt und daher die Anzahl der Partikel, die während eines Zeitschrittes eine Partitionsgränze überqueren, zunimmt. Dies bedeutet, daß das Verhältnis von Kommunikations- zu Rechenzeit mit wachsenden Prozessorzahlen ansteigt.

Der Speedup und die Effizienz für die kollisionsbehafteten Rechnungen im Fall der Endverteilung sind wesentlich niedriger als für alle anderen Testfälle. Dies ist dadurch begründet, daß in diesem Fall die Verteilung der Partikel im Strömungsgebiet sehr ungleichmäßig ist. Zum einen ist, wie bereits in der Abbildung 6.19 ge-

	ohne Kollisionen		mit Kollisionen	
	Startvert.	Endvert.	Startvert.	Endvert.
Anzahl Partikel	10 000	10 000	10 000	10 000
Anzahl Zeitschritte	250	220	230	230
Größe eines Zeitschritts Δt	10^{-4} s	10^{-4} s	10^{-4} s	10^{-4} s

Tabelle 6.6: Simulationsparameter für die parallele Berechnung der horizontalen Kanalströmung unter Verwendung der simultanen Partikelverfolgung

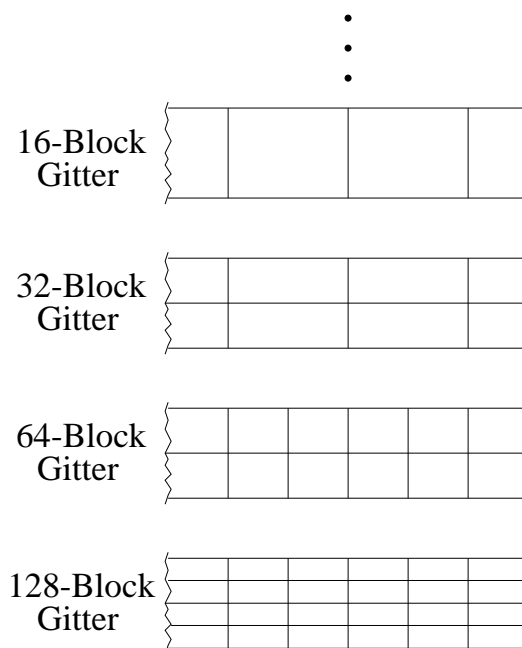


Abbildung 6.26: Prinzip der Gitterpartitionierung für die parallele simultane Partikelverfolgung

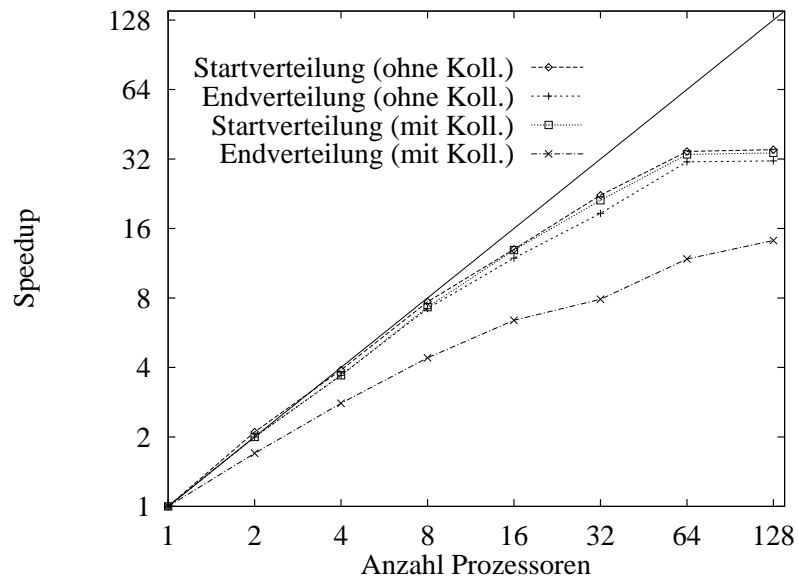


Abbildung 6.27: Speedup für die parallele Simulation auf der Basis der simultanen Partikelverfolgung

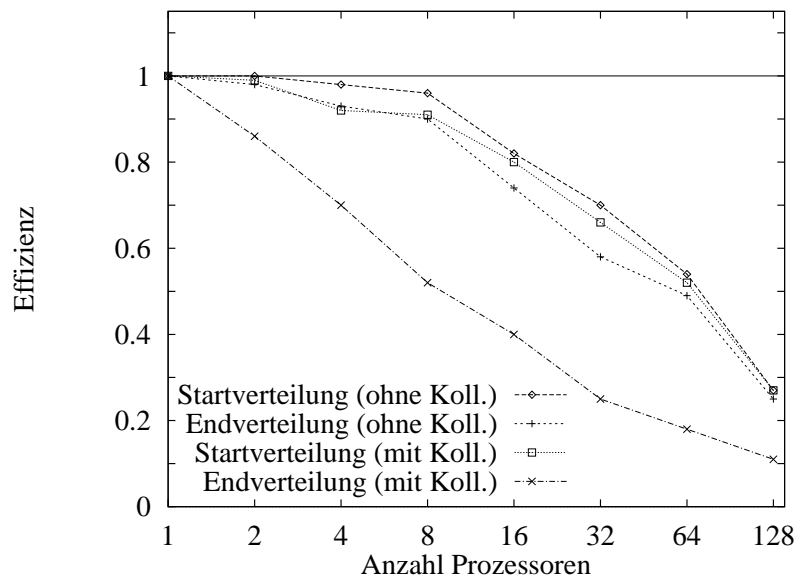


Abbildung 6.28: Effizienz für die parallelen Simulation auf der Basis der simultanen Partikelverfolgung

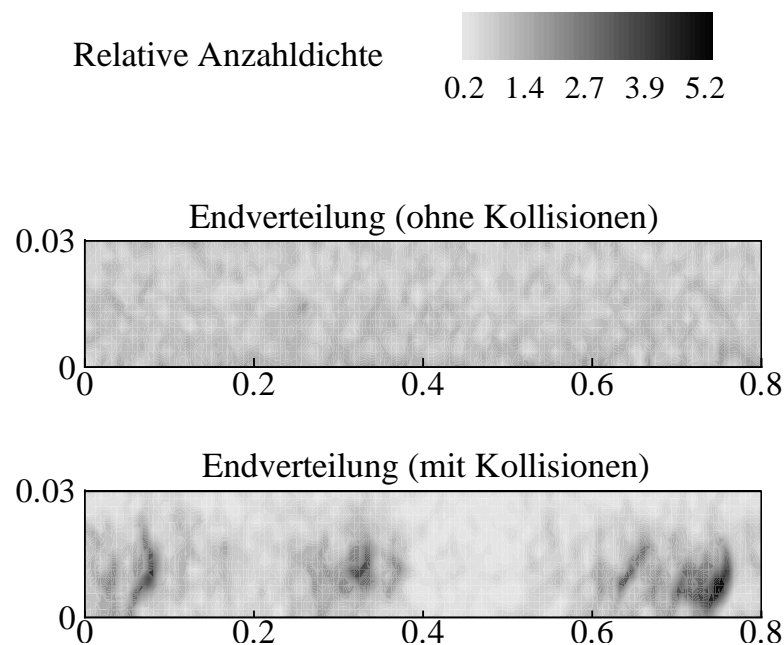


Abbildung 6.29: Endverteilung der Partikel im kollisionsfreien und kollisionsbehafteten Fall

zeigt, die mittlere Partikelkonzentration in der unteren Kanalhälfte größer als in der oberen. Darüber hinaus bewirken die Partikel–Partikel–Kollisionen die Bildung von Partikel–Clustern, was in der Abbildung 6.29 gezeigt ist. In dieser Abbildung ist die Verteilung der relativen Anzahldichte im Kanalsegment für die kollisionsfreie und die kollisionsbehaftete Endverteilung dargestellt. Im kollisionsfreien Fall ist diese Verteilung sehr gleichmäßig, wogegen im kollisionsbehafteten Fall die Konzentration lokal stark variiert. Durch diese großen lokalen Unterschiede in der Partikelkonzentration kommt es bei der parallelen Simulation bereits bei geringen Partitions– bzw. Prozessor–Anzahlen zu großen Rechenlastunterschieden und damit zu einem starken Absinken der Effizienz.

Kapitel 7

Schlußfolgerungen zur Anwendbarkeit der parallelen Algorithmen

7.1 Allgemeines

Die Ergebnisse der im Kapitel 6 beschriebenen Untersuchungen zeigen, daß die Leistungsfähigkeit der verschiedenen parallelen Verfahren sehr unterschiedlich ist. Alle hier untersuchten parallelen Verfahren sind in der Lage, die Zeit, die zur Berechnung einer Gas-Partikel-Strömung benötigt wird, zu verringern. Die Effizienz, mit der dies geschieht, d.h. das Verhältnis von eingesetzten Computerressourcen zur erzielten Rechenzeitverkürzung, kann jedoch von Fall zu Fall stark variieren.

Die Leistungsfähigkeit der einzelnen parallelen Algorithmen ist abhängig vom jeweils zu berechnenden Problem. Einen großen Einfluß haben dabei die physikalischen Eigenschaften der konkret betrachteten Strömung, wie z.B. die Konzentrationsverteilung der Partikel. Diese physikalischen Eigenschaften sind der Hauptgegenstand einer Simulation, d.h. sie sind im allgemeinen vor Beginn der Strömungsberechnung nicht bekannt. Aus diesem Grund ist es sehr schwierig, vor Beginn einer Simulation vorherzusagen, welches parallele Verfahren und welche Anzahl von Prozessoren ein optimales Ergebnis, d.h. eine möglichst kurze Rechenzeit, erzielen wird.

Des weiteren hat die Wahl des numerischen Gitternetzes bzw. die Art der Partitionierung dieses Gitters große Auswirkungen auf die Leistungsfähigkeit der parallelen Algorithmen. Die Art der Partitionierung, die für die parallele Berechnung der Gasphase optimale Ergebnisse liefert, ist in der Regel nicht gleichzeitig die optimale Partitionierung für die Berechnung der Partikelphase. Eine Aussage darüber, welche

Partitionierungsstrategie optimal ist, kann nur getroffen werden, wenn die Physik der zu simulierenden Strömung bekannt ist.

7.2 Parallele Trajektorienberechnung

Domain–Decomposition–Verfahren

Der Hauptvorteil des DD–Verfahrens besteht darin, daß es algorithmisch relativ einfach und daher der Implementierungsaufwand verhältnismäßig gering ist. Des weiteren ist der Kommunikationsaufwand niedrig und die Organisation des Datenaustausches relativ einfach, da ausschließlich eine Host–Node–Kommunikation stattfindet.

In den im Kapitel 6 beschriebenen Untersuchungen zeigte sich, daß ein wesentlicher Nachteil dieses Verfahrens die feste Zuordnung von Gitterpartitionen zu Node–Prozessen ist. Die Partitionierung des Gitters wurde dabei stets so gewählt, daß sie für die parallele Berechnung der Gasströmung eine optimale Rechenlastverteilung ergab. Verwendet man dieselbe Partitionierung für die Berechnung der Bewegung der Partikelphase, so resultiert daraus im allgemeinen eine ungleichmäßige Verteilung der Rechenlast unter den Nodes.

Trotz dieser Nachteile ist festzustellen, daß die Anwendung dieses Verfahrens meistens zu einer signifikanten Verringerung der Rechenzeiten führt. Für viele industrielle Anwendungen ist die Verkürzung der Rechenzeit einer Simulation von z.B. mehreren Tagen auf einige Stunden ein großer Gewinn, auch wenn das dabei verwendete Verfahren nicht mit optimaler Effizienz arbeitet.

Ein potentiell Anwendungsgebiet für das DD–Verfahren sind Parameterstudien zur Optimierung eines Bauteils oder eines Apparates. Hierzu sind in der Regel eine Vielzahl von Simulationen notwendig, wobei sich die physikalischen Eigenschaften der untersuchten Gas–Partikel–Strömung nur wenig verändern. Betrachtet man z.B. die Optimierung eines Zyklons, wie er zur Staubabscheidung aus Gasen verwendet wird, so ist die Verteilung der Partikel innerhalb des Zyklons relativ unabhängig von der Partikelbeladung oder der Einströmgeschwindigkeit des Gas–Partikel–Gemisches. In solchen Fällen ist es denkbar, in einer ersten Simulation zunächst die Partikelverteilung und damit die Verteilung des Rechenaufwandes im Strömungsgebiet zu untersuchen. Auf der Grundlage dieser Ergebnisse kann dann ein Gitternetz bzw. eine Partitionierung konstruiert werden, die eine gleichmäßige Rechenlastverteilung liefert. Mit diesem neu generierten Gitter können dann die Parameterstudien durchgeführt werden.

Betrachtet man die Simulation kollisionsbehafteter Strömungen mit Hilfe der iterierten Trajektorienberechnung, so ist es hierfür schwieriger, eine optimale Gitterparti-

tionierung zu finden, selbst wenn die Eigenschaften der zu berechnenden Strömung bekannt sind. Im ITB-Verfahren wird zunächst ein kollisionsfreier Iterationsschritt ausgeführt, und nach einer gewissen Anzahl von Iterationen wird ein Endzustand erreicht, der dem stationären Zustand der kollisionsbehafteten Strömung entspricht. Dies bedeutet, daß sich die Verteilung der Partikel und damit die Verteilung des Rechenaufwandes im Verlauf der Simulation verändert. Um auch in diesem Fall die Gitterpartitionierung optimal an die Partikelverteilung anzupassen, müßte man in unterschiedlichen Phasen der Berechnung verschiedene Gitternetze verwenden. Ob dieser zusätzliche Aufwand im Vergleich zur erzielbaren Rechenzeitverkürzung lohnenswert ist, läßt sich nur im Einzelfall entscheiden. Des weiteren ist es denkbar, ein Gitternetz zu wählen, das einen Kompromiß darstellt zwischen den Gittern, die jeweils zu Beginn bzw. zum Ende der Simulation optimale Ergebnisse liefern.

Distributed-Shared-Memory-Verfahren

Durch die dynamische Zuordnung von Gitterpartitionen zu Node-Prozessen innerhalb des DSM-Verfahrens wird eine sehr gleichmäßige Verteilung der Rechenlast erreicht. Die Leistungsfähigkeit dieses Verfahrens ist daher, abgesehen von den im Kapitel 6 erwähnten geringen Einschränkungen, nahezu unabhängig von den physikalischen Eigenschaften der betrachteten Strömung oder der Art der Gitterpartitionierung. Auch die Veränderung der Strömungsverhältnisse im Verlauf einer Simulation, wie sie bei der Berechnung kollisionsbehafteter Gas-Partikel-Strömungen mit Hilfe der ITB-Methode auftritt, beeinträchtigt die Leistungsfähigkeit des DSM-Verfahrens nicht.

Diesen Vorteilen steht zur Zeit der Nachteil gegenüber, daß das DSM-Verfahren zur Berechnung der Strömung in N Partitionen insgesamt $2N + 1$ Prozesse benötigt, von denen jedoch nur die N Node-Prozesse die eigentliche Trajektorienberechnung ausführen. Die übrigen $N + 1$ Prozesse dienen dazu, die Berechnung zu optimieren. Viele der heutigen Parallelrechner bieten nicht die Möglichkeit, mehr als einen Prozeß auf einem physischen Prozessor des Rechners zu aktivieren. Dies hat zur Folge, daß zur Strömungsberechnung wesentlich mehr Prozessoren benötigt werden als Gitterpartitionen vorhanden sind, was die parallele Berechnung sehr ineffizient macht. Es ist jedoch zu erwarten, daß zukünftige Rechnergenerationen vermehrt diese Möglichkeit bieten. Des weiteren wird z.B. die standardisierte Message-Passing-Bibliothek MPI in nächster Zukunft die Erzeugung eines Unterprozesses (Thread) innerhalb eines Prozesses und damit die Ausführung unterschiedlicher Aufgaben auf einem Prozessor ermöglichen. Somit könnten z.B. jeweils ein Node- und ein Memory-Manager-Prozeß auf einem gemeinsamen Prozessor aktiv sein, was die Anzahl der benötigten Prozessoren auf $N + 1$ reduzierte. Die gegenseitige Beeinflussung der Prozesse wäre dabei gering, da sie einen komplementären Ressourcenbedarf haben.

Während ein Node relativ viel Rechenleistung beansprucht und wenig Speicherbedarf besitzt, ist es beim Memory-Manager gerade umgekehrt.

7.3 Parallele simultane Partikelverfolgung

Die Berechnungsmethode der simultanen Partikelverfolgung hat im Gegensatz zu der trajektorienbasierten Berechnung den Vorteil, sowohl zur Simulation stationärer als auch instationärer Strömungen geeignet zu sein. Der parallele Algorithmus, der in dieser Arbeit für die simultane Partikelverfolgung angewendet wurde, ist in der Lage, die Rechenzeiten deutlich zu verkürzen. Ähnlich wie das DD-Verfahren hat jedoch auch dieser Algorithmus den Nachteil, daß seine Leistungsfähigkeit aufgrund der statischen Zuordnung von Gitterpartitionen zu Prozessoren stark von der Verteilung der Partikel auf die Partitionen abhängig ist. Diese Partikelverteilung ist einer der Hauptgegenstände der Berechnung und ist im allgemeinen im voraus nicht bekannt. Außerdem kommt hinzu, daß die simultane Partikelverfolgung ein inhärent instationäres Verfahren ist. Die Partikelverteilung im Strömungsgebiet verändert sich daher ständig im Verlauf der Berechnung. Wie im Abschnitt 6.2.2 gezeigt wurde, können selbst in einer Strömung, die im zeitlichen Mittel stationär ist, lokal instationäre und inhomogene Gebiete vorhanden sein. Aus diesen Gründen können mit dem hier verwendeten parallelen Algorithmus im allgemeinen keine optimalen Effizienzen erreicht werden. Allerdings ist, wie bereits im Abschnitt 7.2 erwähnt, in vielen praxisrelevanten Fällen eine deutliche Verringerung der Rechenzeit schon ein großer Gewinn, auch wenn das dabei eingesetzte Verfahren nicht mit höchster Effizienz arbeitet.

Betrachtet man die Simulation von kollisionsfreien Gas-Partikel-Strömungen, so ist eine dynamische Umverteilung der Rechenlast denkbar. Dies könnte in der Weise geschehen, daß die Gitter- und Fluiddaten sowie ein Teil der Partikel von einem Prozessor, der eine Partition mit einer hohen Partikelkonzentration bearbeitet, an einen anderen Prozessor gesendet werden, der momentan eine geringere Auslastung aufweist. Die Bewegung der Partikel in einer Partition würde somit von zwei Prozessoren berechnet. Die Voraussetzung für die Anwendung eines solchen Verfahrens ist, daß der Speicher der Prozessoren groß genug ist, um die Gittergeometrie- und Fluiddaten mehrerer Partitionen aufzunehmen.

Für eine kollisionsbehaftete Simulation ist eine solche dynamische Umverteilung ungleich schwieriger zu realisieren. In diesem Fall werden zur Berechnung der Partikel-Partikel-Kollisionen die Informationen über sämtliche in einer Partition vorhandenen Partikel lokal benötigt. Eine Auslagerung von Teilen der Partikelberechnung auf einen anderen Prozessor wäre deshalb algorithmisch sehr aufwendig und mit einem sehr großen Kommunikationsaufwand verbunden. Einen möglichen Ausweg,

um auch für kollisionsbehaftete Simulationen den Rechenaufwand dynamisch neu verteilen zu können, bietet die Verwendung unstrukturierter Gitternetze. Bei Verwendung solcher Gitter ist die Repartitionierung und der Austausch von Teilen des Gitters mit allen zugehörigen Informationen wesentlich einfacher als bei Verwendung blockstrukturierter Netze, wie in dieser Arbeit geschehen.

Kapitel 8

Zusammenfassung und Ausblick

Im Rahmen der vorliegenden Arbeit wurden Algorithmen zur Berechnung kollisionsbehafteter Gas-Partikel-Strömungen auf massiv parallelen Rechnern entwickelt. Die Leistungsfähigkeit dieser Algorithmen im Hinblick auf die damit zu erzielende Verkürzung der Rechenzeit wurde untersucht. Auf dieser Grundlage wurden Vorschläge für mögliche weitere Verbesserungen der Verfahren bzw. für alternative Parallelisierungsverfahren gemacht.

Die numerische Berechnung der Gasströmung erfolgte auf der Grundlage einer Diskretisierung der Erhaltungsgleichungen nach der Finite-Volumen-Methode für randangepaßte, nichtorthogonale, blockstrukturierte numerische Gitternetze. Zur Lösung der resultierenden linearen Gleichungssysteme für die einzelnen Variablen wurde die SIP-Methode verwendet, und die Lösung des Gesamtsystems unter Berücksichtigung der Druck-Geschwindigkeits-Kopplung erfolgte mit Hilfe eines SIMPLE-Verfahrens.

Zur Berechnung der Partikelbewegung wurde in dieser Arbeit das Lagrange-Verfahren verwendet. Bei den hier betrachteten Gas-Feststoff-Strömungen waren die wichtigsten auf ein Einzelpartikel einwirkenden Kräfte die aerodynamische Widerstandskraft, die Magnus-Kraft, die Saffman-Kraft und die Gravitationskraft. Die makroskopischen Strömungsgrößen der Partikelphase wurden auf zwei verschiedene Arten bestimmt, mit der Methode der Trajektorienberechnung sowie mit Hilfe der simultanen Partikelverfolgung. Zur Berücksichtigung der Partikel-Partikel-Kollisionen innerhalb der trajektorienbasierten Simulation wurde ein iteratives Verfahren nach Oesterlé und Petitjean[34] verwendet. Im Rahmen der simultanen Partikelverfolgung wurden die Kollisionen unter Verwendung einer Monte-Carlo-Methode berechnet, wie sie auch bei der Simulation verdünnter Gasströmungen angewendet wird.

Die parallele Berechnung der Gasströmung erfolgte auf der Basis eines Gebietszerlegungsverfahrens. Dabei wurde das numerische Gitternetz in eine bestimmte Anzahl

gleichgroßer Partitionen aufgeteilt. Die Strömung in jedem dieser Teilgebiete wurde von einem separaten Prozessor des Parallelrechners berechnet.

Für die parallele Simulation der kollisionsbehafteten Partikelbewegung auf der Basis der Trajektorienberechnung wurden zwei bereits existierende Algorithmen unter Einbeziehung der Kollisionen weiterentwickelt. Das Domain-Decomposition- (DD) Verfahren beruht auf einer statischen Zuordnung von Gitterpartitionen zu Prozessen. Alle Trajektoriensegmente, die eine bestimmte Partition durchqueren, werden dabei von dem Prozeß berechnet, dem diese Partition zugeordnet ist. Im Gegensatz dazu erfolgt beim Distributed-Shared-Memory- (DSM) Verfahren die Zuordnung von Gitterpartitionen zu Prozessen auf dynamische Weise. Dadurch können die ein bestimmtes Teilgebiet durchquerenden Trajektorienabschnitte von verschiedenen Prozessen berechnet werden. Für die dynamische Neuverteilung der Partitionen werden zusätzliche Prozesse benötigt, so daß die Gesamtzahl der Prozesse im DSM-Verfahren wesentlich größer als im DD-Verfahren ist.

Zur Parallelisierung der simultanen Partikelverfolgung wurde ein neuer Algorithmus entwickelt, bei dem, ähnlich wie beim DD-Verfahren, einem Prozessor des Parallelrechners jeweils ein Teilgebiet der Strömungsgeometrie statisch zugewiesen wird. Die Bewegung aller Partikel die sich momentan in diesem Teilgebiet befinden, wird von diesem Prozessor berechnet. Erreicht ein Partikel eine Partitions-grenze, so werden die zugehörigen Partikeldaten an den Prozessor übertragen, dem die entsprechende Nachbarpartition zugeordnet ist.

Zur Untersuchung der Leistungsfähigkeit der parallelen Algorithmen wurden Gas-Partikel-Strömungen in zwei verschiedenen zweidimensionalen Testgeometrien betrachtet, einem vertikalen und einem horizontalen Kanal. Dazu wurden zunächst die physikalischen Strömungseigenschaften untersucht, wobei jeweils eine niedrig beladene, kollisionsfreie Strömung und eine höher beladene, kollisionsbehaftete Strömung berechnet wurde. Die Ergebnisse dieser Simulationen zeigten eine relativ gute qualitative Übereinstimmung mit Ergebnissen aus der Literatur für dreidimensionale Rohrströmungen. Insbesondere zeigte sich, daß der Einfluß der Partikel-Partikel-Kollisionen bei den höher beladenen Strömungen eine deutliche Veränderung des Konzentrations- und Geschwindigkeitsprofils der Partikelphase im Vergleich zur jeweiligen kollisionsfreien Strömung zur Folge hat.

Bei der Untersuchung der parallelen Algorithmen zeigte sich, daß in allen Fällen die physikalischen Strömungsverhältnisse sowie die Art der Gitterpartitionierung die Effizienz der Verfahren stark beeinflussen. Beim DD-Verfahren zur parallelen Trajektorienberechnung ist dieser Einfluß wesentlich stärker zu beobachten als beim DSM-Verfahren. Da beim DD-Verfahren die Zuordnung von Partitionen zu Prozessen statisch ist, hängt die Leistungsfähigkeit dieses Verfahrens wesentlich davon ab, wie gut der Gesamtrechnaufwand durch die gewählte Art der Partitionierung

skaliert werden konnte. Eine solche Abhängigkeit ist zwar auch beim DSM-Verfahren zu beobachten, allerdings ist hier durch die Möglichkeit der dynamischen Zuweisung von Partitionen der Rechenaufwand insgesamt wesentlich gleichmäßiger unter den Prozessen verteilt. Mit dem DSM-Verfahren wurden deshalb höhere Effizienzen erzielt als mit dem DD-Verfahren.

Eine Möglichkeit, das DD-Verfahren effizient anzuwenden, bietet sich bei Parameterstudien zur Optimierung von Bauteilen oder Apparaten. Hierbei ist in der Regel das Strömungsverhalten bekannt, das sich durch die Veränderung einzelner Parameter nicht wesentlich verändert. Mit dieser Kenntnis kann vor Beginn einer Serie von Simulationen ein Gitternetz bzw. eine Partitionierung konstruiert werden, mit der eine gleichmäßige Verteilung des Rechenaufwandes erreicht wird.

Das DSM-Verfahren erzielt unabhängig vom konkreten Anwendungsfall stets eine sehr gleichmäßige Rechenlastverteilung. Der derzeitige Nachteil dieses Verfahrens besteht in der großen Anzahl von Prozessen, die dazu benötigt werden. Dieses Verfahren kann nur dann effizient eingesetzt werden, wenn der verwendete Parallelrechner bzw. die verwendete Message-Passing-Bibliothek die Aktivierung mehrerer Prozesse auf einem Prozessor des Rechners zulassen. Aufgrund der derzeitigen Entwicklungen im Bereich der Betriebssysteme und der Message-Passing-Bibliotheken ist abzusehen, daß diese Möglichkeit in nächster Zukunft vermehrt gegeben sein wird.

Bei der parallelen simultanen Partikelverfolgung ist, ähnlich wie beim DD-Verfahren, ebenfalls eine starke Abhängigkeit der Effizienz von der Verteilung des Rechenaufwandes auf die Gitterpartitionen zu beobachten. Dabei hängt die Größe des Rechenaufwandes, der in einer Partition anfällt, hauptsächlich von der Anzahl der Partikel ab, die sich in dieser Partition befinden. Da die simultane Partikelverfolgung ein instationäres Berechnungsverfahren ist, verändert sich diese Anzahl permanent im Verlauf der Simulation. Infolgedessen verändert sich auch die Effizienz des parallelen Algorithmus während der Berechnung, falls die Partitionierung und die Zuordnung von Partitionen zu den Prozessoren des Parallelrechners statisch durchgeführt wird.

Eine Steigerung der Effizienz der parallelen simultanen Partikelverfolgung scheint aus heutiger Sicht nur durch eine dynamische Umverteilung der Rechenlast während der Simulation möglich. Die Verwendung unstrukturierter Gitternetze wäre hierbei von Vorteil, da in diesem Fall die Repartitionierung des Gitters einfacher und der notwendige Kommunikationsaufwand geringer wäre als bei den in dieser Arbeit verwendeten blockstrukturierten Gitternetzen.

Zusammenfassend kann gesagt werden, daß alle hier vorgestellten parallelen Verfahren prinzipiell für den Einsatz auf massiv parallelen Rechnern geeignet sind. Durch die Verwendung dieser Verfahren können die Rechenzeiten für Simulationen von

Gas-Partikel-Strömungen deutlich verringert werden. Insbesondere sind dadurch die Voraussetzungen gegeben, kollisionsbehaftete Strömungen in komplexen dreidimensionalen Strömungsgebieten in einer angemessenen Zeit zu berechnen. Jedoch sind die verschiedenen Verfahren nicht universell und mit gleicher Effizienz für alle Anwendungsfälle einsetzbar, und teilweise ist eine Weiterentwicklung zur Steigerung der Effizienz erforderlich.

Anhang A

Erläuterungen zur Diskretisierung und Lösung der Erhaltungsgleichungen für die Gasströmung

In diesem Anhang sind einige Herleitungen und Verfahren zusammengefaßt, die für die im Kapitel 2 beschriebene Diskretisierung und numerische Lösung der Bilanzgleichungen benötigt werden. Die Darstellungen der folgenden Abschnitte sind eine Kurzfassung der ausführlichen Darstellungen in [36].

A.1 Approximation der diffusiven Flüsse

Der in den Bilanzgleichungen auftretende Diffusionsterm F^d hat die allgemeine Form:

$$F^d = \int_S \Gamma \operatorname{grad} \phi \cdot \vec{n} \, dS . \quad (\text{A.1})$$

Der diffusive Fluß durch die Seitenfläche ϵ wird approximiert durch:

$$F_\epsilon^d = \int_{S_\epsilon} \Gamma \operatorname{grad} \phi \cdot \vec{n} \, dS \approx (\Gamma \operatorname{grad} \phi \cdot \vec{n})_\epsilon S_\epsilon . \quad (\text{A.2})$$

Der Gradient von ϕ im Mittelpunkt der Fläche kann sowohl in globalen kartesischen Koordinaten (x, y) als auch durch die lokalen orthogonalen Koordinaten (n, t) ausgedrückt werden:

$$\operatorname{grad} \phi = \left(\frac{\partial \phi}{\partial x}, \frac{\partial \phi}{\partial y} \right)^T = \frac{\partial \phi}{\partial n} \vec{n} + \frac{\partial \phi}{\partial t} \vec{t} , \quad (\text{A.3})$$

wobei n die Koordinatenrichtung senkrecht zur Seitenfläche und t die tangentielle Koordinatenrichtung ist (siehe Abbildung 2.2). Setzt man diese Formulierung für den Gradienten in die Gleichung (A.2) ein, so sieht man, daß nur die Ableitung senkrecht zur Seitenfläche einen Beitrag zum diffusiven Fluß liefert, d.h.:

$$F_e^d \approx \Gamma_e \left(\frac{\partial \phi}{\partial n} \right)_e S_e \quad (\text{A.4})$$

Für orthogonale Gitternetze kann die Ableitung senkrecht zur Fläche auf einfache Weise mit einer zentralen Differenz berechnet werden:

$$\left(\frac{\partial \phi}{\partial n} \right)_e \approx \frac{\phi_E - \phi_P}{L_{P,E}}, \quad (\text{A.5})$$

wobei $L_{P,E}$ der Abstand der Punkte P und E ist. In nichtorthogonalen Gittern ist die Verbindungslinie zwischen den benachbarten Gitterpunkten nicht senkrecht zur dazwischenliegenden Seitenfläche. Für diesen Fall wird die folgende Deferred-Correction Formulierung verwendet:

$$\left(\frac{\partial \phi}{\partial n} \right)_e \approx \left(\frac{\partial \phi}{\partial \xi} \right)_e^{neu} + \left[\left(\frac{\partial \phi}{\partial n} \right)_e^{alt} - \left(\frac{\partial \phi}{\partial \xi} \right)_e^{alt} \right]. \quad (\text{A.6})$$

Hierbei ist ξ die Koordinatenrichtung entlang der Verbindungslinie der Punkte P und E . Der mit dem Superskript neu gekennzeichnete erste Term auf der rechten Seite wird durch einen impliziten Ausdruck berechnet:

$$\left(\frac{\partial \phi}{\partial \xi} \right)_e^{neu} = \frac{(\phi_E - \phi_P)^{neu}}{L_{P,E}}, \quad (\text{A.7})$$

und die mit dem Superskript alt gekennzeichneten Terme werden auf explizite Weise ermittelt:

$$\left(\frac{\partial \phi}{\partial n} \right)_e^{alt} = (\text{grad } \phi)_e^{alt} \cdot \vec{n}; \quad \left(\frac{\partial \phi}{\partial \xi} \right)_e^{alt} = (\text{grad } \phi)_e^{alt} \cdot \vec{i}_\xi, \quad (\text{A.8})$$

wobei \vec{i}_ξ der Koordinatenvektor der ξ -Richtung ist. Nach dem Einsetzen der Gleichungen (A.6)–(A.8) in die Gleichung (A.4) ergibt sich als Approximation für den diffusiven Fluß durch die Seitenfläche e :

$$F_e^d \approx \Gamma_e \frac{S_e}{L_{P,E}} (\phi_E - \phi_P)^{neu} + \Gamma_e S_e (\text{grad } \phi)_e^{alt} \cdot (\vec{n} - \vec{i}_\xi). \quad (\text{A.9})$$

A.2 Berechnung der Ableitung im Zellenmittelpunkt

Die Berechnung der Ableitung einer Größe ϕ im Mittelpunkt einer Gitterzelle soll im folgenden am Beispiel der Ableitung nach der Variablen x erläutert werden. Die Berechnung der Ableitung nach y erfolgt in analoger Weise. Zunächst wird die Ableitung im Mittelpunkt der Zelle durch den Mittelwert der Ableitung in der Zelle approximiert:

$$\left(\frac{\partial \phi}{\partial x}\right)_P = \frac{\int_{\Omega} \frac{\partial \phi}{\partial x} d\Omega}{\Delta \Omega}. \quad (\text{A.10})$$

Die Ableitung $\partial \phi / \partial x$ kann als Divergenz des Vektors $(\phi, 0)^T$ betrachtet werden. Damit kann das Volumenintegral in Gleichung (A.10) unter Anwendung des Gaußschen Integralsatzes in ein Oberflächenintegral umgewandelt werden:

$$\int_{\Omega} \frac{\partial \phi}{\partial x} d\Omega = \int_S (\phi, 0)^T \cdot \vec{n} dS \approx \sum_c \phi_c S_{cx}, \quad c = e, n, w, s. \quad (\text{A.11})$$

Setzt man diese Beziehung in die Gleichung (A.10) ein, so erhält man als Approximation für die Ableitung im Zellenmittelpunkt:

$$\left(\frac{\partial \phi}{\partial x}\right)_P \approx \frac{\sum_c \phi_c S_{cx}}{\Delta \Omega}. \quad (\text{A.12})$$

A.3 Lösung des linearen Gleichungssystems für eine Strömungsvariable mit der SIP-Methode

Das linearisierte Gleichungssystem für eine Strömungsvariable ϕ läßt sich in der Matrixschreibweise darstellen als:

$$\overline{\overline{A}} \vec{\phi} = \vec{Q}. \quad (\text{A.13})$$

Hierbei enthält der Vektor $\vec{\phi}$ die unbekannten Werte der Variablen ϕ in allen Zellenmittelpunkten, die Matrix $\overline{\overline{A}}$ enthält die Koeffizienten der Variablen und im Quelltermvektor \vec{Q} sind alle bekannten Größen der Gleichung zusammengefaßt. Betrachtet man einen iterativen Lösungsprozeß, so erhält man nach n Iterationen eine approximative Lösung $\vec{\phi}^n$, die die Gleichung (A.13) nicht exakt erfüllt:

$$\overline{\overline{A}} \vec{\phi}^n = \vec{Q} - \vec{r}^n, \quad (\text{A.14})$$

Hierin ist $\vec{\rho}^n$ der Residuenvektor, der im Laufe des Iterationsprozesses zu Null werden soll. Ein allgemeines iteratives Lösungsschema für die Gleichung (A.13) ist z.B. gegeben durch:

$$\overline{\overline{M}} \vec{\phi}^{n+1} = \overline{\overline{N}} \vec{\phi}^n + \vec{B} . \quad (\text{A.15})$$

Im Konvergenzfall ist $\vec{\phi}^{n+1} = \vec{\phi}^n = \vec{\phi}$. Daraus folgt, daß für diesen Fall gelten muß:

$$\overline{\overline{A}} = \overline{\overline{M}} - \overline{\overline{N}} \quad \text{und} \quad \vec{B} = \vec{Q} . \quad (\text{A.16})$$

Subtrahiert man nun den Term $\overline{\overline{M}} \vec{\phi}^n$ von der Gleichung (A.15), so ergibt sich unter Berücksichtigung der Gleichung (A.14) die Beziehung:

$$\overline{\overline{M}} \vec{\delta}^n = \vec{\rho}^n , \quad (\text{A.17})$$

wobei $\vec{\delta}^n = \vec{\phi}^{n+1} - \vec{\phi}^n$ der Korrekturterm ist, der die Änderung von $\vec{\phi}$ von einer inneren Iteration zur nächsten beschreibt. Die Lösung dieser Gleichung erfolgt mit Hilfe der SIP-Methode (Strongly Implicit Procedure) nach Stone [44]. Dabei wird die Matrix $\overline{\overline{M}}$ als Produkt aus einer unteren und einer oberen Dreiecksmatrix dargestellt:

$$\overline{\overline{M}} = \overline{\overline{L}} \overline{\overline{U}} . \quad (\text{A.18})$$

Die Koeffizienten von $\overline{\overline{L}}$ und $\overline{\overline{U}}$ ergeben sich aus der Koeffizientenmatrix $\overline{\overline{A}}$. Eine ausführliche Beschreibung der Koeffizientenberechnung ist z.B. in [36] gegeben. Der Zusammenhang zwischen dem Korrekturterm und dem Residuum stellt sich nun wie folgt dar:

$$\overline{\overline{L}} \overline{\overline{U}} \vec{\delta}^{n+1} = \vec{\rho}^n . \quad (\text{A.19})$$

Nach der Berechnung von $\overline{\overline{L}}$ und $\overline{\overline{U}}$ läßt sich nun der Korrekturterm auf einfache Weise in drei Schritten bestimmen:

$$\vec{\rho}^n = \vec{Q} - \overline{\overline{A}} \vec{\phi}^n , \quad (\text{A.20})$$

$$\vec{R}^n = \overline{\overline{L}}^{-1} \vec{\rho}^n , \quad (\text{A.21})$$

$$\vec{\delta}^{n+1} = \overline{\overline{U}}^{-1} \vec{R}^n . \quad (\text{A.22})$$

Der aktualisierte Wert für die Variable ϕ ergibt sich aus dem Wert der vorherigen Iteration und dem Korrekturterm zu:

$$\vec{\phi}^{n+1} = \vec{\phi}^n + c_{urf} \vec{\delta}^{n+1} , \quad (\text{A.23})$$

wobei c_{urf} ein Unterrelaxationsfaktor mit $0 < c_{urf} < 1$ ist.

Es ist anzumerken, daß die SIP-Methode kein allgemeines Lösungsschema für beliebige Gleichungssysteme darstellt. Die Anwendung dieser Methode setzt voraus, daß die Lösung glatt ist. Dies kann im vorliegenden Fall angenommen werden, da die zu lösenden Gleichungen Approximationen von elliptischen partiellen Differentialgleichungen sind, die eine stationäre, inkompressible Strömung beschreiben.

Literaturverzeichnis

- [1] Basset, A.B.: A treatise on Hydrodynamics, Cambridge University Press, Cambridge, 1888
- [2] Bird, G.A.: Molecular gas dynamics, Oxford University Press, 1976
- [3] Bird, G.A.: Molecular gas dynamics and the direct simulation of gas flows, Clarendon Press, Oxford, 1994
- [4] Boussinesq, J.: Sur la résistance qu'oppose ... soient négligeable, C. R. Acad. Sci. Paris, Vol. 100, 1885, S. 935–937
- [5] Carretto, L.S., Gosman, A.D., Patankar, S.V., Spalding, D.B.: Two calculation procedures for steady, three-dimensional flows with recirculation, Proc. 3rd Int. Conf. Num. Methods Fluid Dyn., Paris, 1972, Vol. II, S. 60
- [6] Clift, R., Grace, J.R., Weber, M.E.: Bubbles, drops and particles, Academic Press, New York, San Francisco, London, 1978
- [7] Crowe, C.T., Sharma, M.P., Stock, D.E.: The particle-source-in cell (PSI-CELL) model for gas-droplet flows, J. Fluids Eng., Vol. 99, 1977, S. 325–332
- [8] Crowe, C.T.: On the relative importance of particle-particle collisions in gas-particle flows, Proc. Conf. on Gas Borne Particles, Paper C78/81, 1981, S. 135–137
- [9] Crowe, C.T.: REVIEW-Numerical models for dilute gas-particle flows, J. Fluids Eng., Vol. 104, 1982, S. 297–303
- [10] Crowe, C.T.: The state-of-the-art in the development of numerical models for dispersed phase flows, Proc. The International Conference on Multiphase Flows, Tsukuba, Japan, 1991, Vol. 3, S. 49–60
- [11] Dandy, D.S., Dwyer, H.A.: A sphere in shear flow at finite Reynolds number: Effect of shear on particle lift, drag and heat transfer, J. Fluid Mech., Vol. 216, 1990, S. 381–410

- [12] Dennis, S.C.R., Singh, S.N., Ingham, D.B.: The steady flow due to a rotating sphere at low and moderate Reynolds numbers, *J. Fluid Mech.*, Vol. 101, 1980, S. 257–279
- [13] Dongarra, J.J., Meuer, H.W., Strohmaier, E.: TOP500 Supercomputer Sites, 11th Edition, June 18, 1998
- [14] Elghobashi, S.E., Abou–Arab, T.W.: A two–equation turbulence model for two–phase flows, *Phys. Fluids*, Vol. 26, 1983, S. 931–938
- [15] Frank, Th.: Numerische Berechnung der feststoffbeladenen Gasströmung im horizontalen Kanal unter Berücksichtigung von Wandrauhigkeiten, Dissertation, Bergakademie Freiberg, 1992
- [16] Frank, Th.: Comparison of three parallelization methods for calculation of disperse multiphase flows using the Lagrangian approach, *Proc. 3rd Int. Conf. Parallel CFD '96, Implementations and Results Using Parallel Computers*, Capri, Italy, May 20–23, 1996
- [17] Frank, Th., Wassen, E.: Parallel solution algorithms for Lagrangian simulation of disperse multiphase flows, *Proc. 2nd Int. Symposium on Numerical Methods for Multiphase Flows*, ASME Fluids Engineering Division Summer Meeting, San Diego, CA, USA, July 7–11, 1997, FED–Vol. 236, Vol. 1, S. 11–20
- [18] Frank, Th., Wassen, E.: Parallel efficiency of PVM- and MPI-implementations of two algorithms for the Lagrangian prediction of disperse multi–phase flows, *JSME Centennial Congress 1997, ISAC '97 Conference on Advanced Computing on Multiphase Flow*, Tokyo, Japan, July 18–19, 1997
- [19] Frank, Th., Wassen, E., Yu, Q.: A 3–dimensional Lagrangian solver for disperse multiphase flows on arbitrary, geometrically complex flow domains using block-structured numerical grids, *7th Int. Symposium on Gas–Particle Flows*, ASME Fluids Engineering Division Summer Meeting, Vancouver, BC, Kanada, June 22–26, 1997, CD–ROM Proceedings, Paper No. FEDSM97–3590
- [20] Haan, O.: Vektorrechner: Architektur – Programmierung – Anwendung, K.G. Saur, München, 1993
- [21] Huber, N.: Zur Phasenverteilung von Gas–Feststoff–Strömungen in Rohren, Dissertation, Erlangen, 1997
- [22] Illner, R., Neunzert, H.: On simulation methods for the Boltzmann equation, *Transport Theory and Statistical Physics*, Vol. 16, 1987, S. 141–154

- [23] Khosla, P.K., Rubin, S.G.: A diagonally dominant second-order accurate implicit scheme, *Computers Fluids*, Vol. 2, 1974, S. 207–209
- [24] Kitron, A., Elperin, T., Tamir, A.: Monte Carlo analysis of wall erosion and direct contact heat transfer by impinging two-phase jets, *Journal of Thermophysics and Heat Transfer*, Vol. 3, No. 2, 1989, S. 112–122
- [25] Kitron, A., Elperin, T., Tamir, A.: Monte Carlo simulation of gas–solids suspension flows in impinging stream reactors, *Int. J. Multiphase Flow*, Vol. 16, No. 1, 1990, S. 1–17
- [26] Koura, K.: Null-collision technique in the direct-simulation Monte Carlo method, *Phys. Fluids*, Vol. 29, No. 11, 1986, S. 3509–3511
- [27] Matsumoto, S., Saito, S.: On the mechanism of suspension of particles in horizontal pneumatic conveying: Monte-Carlo simulation based on the irregular bouncing model, *J. Chemical Eng. of Japan*, Vol. 9, No. 1, 1970, S. 83–92
- [28] Matsumoto, S., Saito, S.: Monte Carlo simulation of horizontal pneumatic conveying based on the rough wall model, *J. Chemical Eng. of Japan*, Vol. 3, No. 1, 1970, S. 223–230
- [29] Matsumoto, S., Saito, S., Maeda, S.: Simulation of gas–solid two-phase flow in horizontal pipe, *J. Chemical Eng. of Japan*, Vol. 9, No. 1, 1976, S. 23–28
- [30] Mei, R.: An approximate expression for the shear lift force on a spherical particle at finite Reynolds number, *Int. J. Multiphase Flow*, Vol. 18, 1992, S. 145–147
- [31] Milojević, D.: Lagrangian stochastic–deterministic (LSD) prediction of particle dispersion in turbulence, *Particle and Particle Systems Characterization*, Vol. 7, 1990, S. 181–190
- [32] Morikawa, Y., Tsuji, Y., Tanaka, T.: Measurements of horizontal air–solid two-phase flow using an optical fiber probe, *Bull JSME*, Vol. 29, 1986, S. 802–809
- [33] Morsi, S.A., Alexander, A.J.: An investigation of particle trajectories in two-phase flow systems, *J. Fluid Mech.* Vol. 55, No. 2, 1972, S. 193–208
- [34] Oesterlé, B., Petitjean, A.: Simulation of particle-to-particle interactions in gas–solid flows, *Int. J. Multiphase Flow*, Vol. 19, No. 1, 1993, S. 199–211
- [35] Patankar, S.V., Spalding, D.B.: A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, *Int. J. Heat Mass Transfer*, Vol. 15, 1972, S. 1787

- [36] Ferziger, J.H., Perić, M.: Computational methods for fluid dynamics, Springer-Verlag, 1996
- [37] Reif: Fundamentals of Statistics and Thermal Physics, McGraw-Hill, 1965
- [38] Saffman, P.G.: The lift on a small sphere in a shear flow, J. Fluid Mech., Vol. 22, 1965, S. 385–400
- [39] Sakiz, M., Simonin, O.: Continuum modelling and Lagrangian simulation of the turbulent transport of kinetic particle stresses in a vertical gas–solid channel flow, 3rd Int. Conf. on Multiphase Flow ICMF’98, Lyon, France, June 8–12, 1998, CD-ROM Proceedings, Paper No. 537
- [40] Shuen, J.-S., Solomon, A.S.P., Zhang, Q.-F., Faeth, G.M.: A theoretical and experimental study of turbulent particle-laden jets, NASA Contractor Report 168293, 1983
- [41] Sommerfeld, M.: Modelling of particle-wall collisions in confined gas–particle flows, Int. J. Multiphase Flow, Vol. 18, No. 6, 1992, S. 905–926
- [42] Sommerfeld, M.: A stochastic approach to model inter-particle collisions in the frame of the Euler/Lagrange approach, ERCOFTAC, Bulletin, March 1998
- [43] Squires, K.D., Eaton, J.K.: On the modeling of particle-laden turbulent flows, Proc. 6th Workshop on Two-phase Flow Predictions, edited by M. Sommerfeld, Bilateral Seminars of the International Bureau, Forschungszentrum Jülich GmbH, Vol. 14, 1993, S. 220–229
- [44] Stone, H.L.: Iterative solution of implicit approximations of multidimensional partial differential equations, SIAM J. Numer. Anal., Vol. 5, 1968, S. 530–558
- [45] Tanaka, T.: Measurements of gas–solid two-phase flow in a vertical pipe, Transactions of JSME, Vol. 55, No. 516, B, S. 2302–2309
- [46] Tanaka, T., Tsuji, Y.: Numerical simulation of gas–solid two-phase flow in a vertical pipe: On the effect of inter-particle collision, ASME FED–Vol. 121, Gas–Solid Flows, Book No. G00609, 1991, S. 123–128
- [47] Tanaka, T., Kiribayashi, K., Tsuji, Y.: Monte Carlo simulation of gas–solid flow in vertical pipe or channel, Proc. Int. Conf. on Multiphase Flows ’91–Tsukuba, September 24–27, 1991, Tsukuba, Japan, S. 439–442
- [48] Tashiro, H., Tomita, Y., Peng, X., Kishikawa, Y.: A numerical simulation of a gas–solid two-phase upward flow in a vertical pipe, Proc. Int. Conf. on Multiphase Flows ’91, Tsukuba, Japan, September 24–27, 1991, S. 435–438

- [49] Tsuji, Y., Morikawa, Y., Mizuno, O.: Experimental measurement of the Magnus-force on a rotating sphere at low Reynolds numbers, Transactions of ASME, J. Fluids Eng., Vol. 107, 1981, S. 484–488
- [50] Tsuji, Y., Oshima, T., Morikawa, Y.: Numerical simulation of pneumatic conveying in a horizontal pipe, KONA–Powder Science and Technology in Japan, Vol. 3, 1985, S. 38–51
- [51] Tsuji, Y.: Discrete element modelling of Clusters in gas–solid flows, Proc. 2nd Int. Symposium on Numerical Methods for Multiphase Flows, ASME Fluids Engineering Division Summer Meeting, San Diego, CA, USA, July 7–11, 1996, FED–Vol. 1, S. 3
- [52] Tysinger, T.L., Missaghi, M.: A combined shared–memory and distributed–memory model for computation of coupled Lagrangian dispersed phase and Eulerian gas phase combustion, Proc. Int. Conf. on Recent Developments and Advances using Parallel Computers, Parallel CFD '97, Manchester, England, May 19–21, 1997
- [53] Umois, H., Kleis, I.: A critical analysis of erosion problems which have been little studied, Wear, Vol. 31, 1975, S. 359–371
- [54] Vojir, D.J., Michaelides, E.E.: Effect of the history term on the motion of rigid spheres in a viscous fluid, Int. J. Multiphase Flow, Vol. 20, No. 3, 1994, S. 547–556
- [55] Wassen, E., Frank, Th., Yu, Q.: A comparison of parallel algorithms for the numerical simulation of multiphase flows, Advances in Computational Mechanics with Parallel and Distributed Computing, Civil–Comp Press, 1997, S. 179–185, Proc. 1st Euro-Conference on Parallel and Distributed Computing for Computational Mechanics, Lochinver, Schottland, April 26 – Mai 1, 1997
- [56] Yonemura, S., Tanaka, T., Tsuji, Y.: Cluster formation in gas–solid flow predicted by the DSMC method, ASME FED–Vol. 166, Gas–Solid Flows, Book No. H00806, 1993, S. 303–309